RAB

PROCEEDINGS OF THE FIFTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 5th '00)

Vol.1

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

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

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

I

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 ARTI-
FICIAL LIFE AND ROBOTICS (Springer) and APPLIED MATHE-
MATICS AND COMPUTATION (North-Holland).

All correspondence relating to the symposium
should be addressed to:

MESSAGE

Masanori Sugisaka
General Chairman of AROB
(Professor, Oita University)
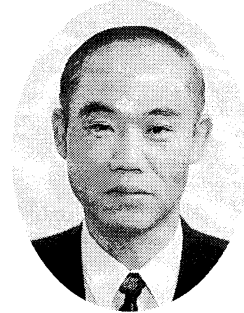
It is my great honor to invite you all to The Fifth International Symposium on Artificial Life and Robotics(AROB 5th '00), organized by Oita University under the sponsorship of Ministry of Education, Science, Sports, and Culture(Monbusho), Japanese Government and co-sponsored by Santa Fe Institute(SFI), 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.
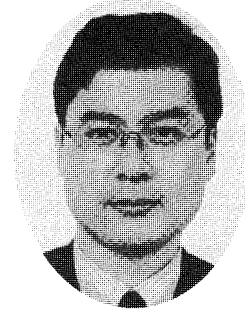
I am looking forward to meeting you in Oita.

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.
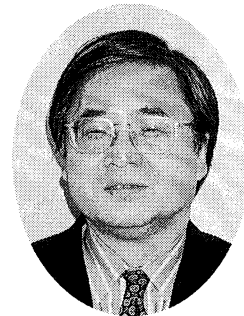
J.L. Casti

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.

H. Tanaka

January 12, 2000

TECHNICAL PAPER INDEX

Plenary Lectures (Plenary Talks)

Banquet Talk

Invited Lectures (Invited Talks)

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

INTELLIGENT SYSTEMS
- A New Horizon of Science and Technology -

Fumio Harashima

Tokyo Metropolitan Institute of Technology

20th century, a century of war and economical growth is over. What is the major issue for human society in the coming century? The most critical issue is "Human Survival" in relation with global environmental pollution. It will take 30-50 years from now before human beings would be convinced of their survival in the future. The human survival is not the final goal of the society, but it is just a process of the future of human society.

What kind of human society can we expect in the middle of 21st century. My prediction is,

"A society which enhances human intelligence by the use of science and technology"

In this presentation, first, the history and present status of science and technology are reviewed, and then a new direction of science and technology is suggested. A new engineering discipline called "Intelligent Systems" will be the key technology to realize "Intelligent Human Lives". ARTIFICIAL LIFE AND ROBOTICS, which is the topic of this Conference will play a major role in this new direction of science and technology.

Several projects have already started toward Intelligent Human Society. They are,

- IMS
- Electric/Hybrid Vehicles
- ITS
- Disaster Prevention System
- Distance Learning

Some videos are shown in this presentation.

Brain-Style Information Systems --- Mathematical Foundations

Shun-ichi Amari

RIKEN Brain Science Institute

Abstract

The brain processes information by using parallel dynamics of neural excitation over enormous number of neurons. It has, moreover, an ability to learn from examples and to self-organize its structures and functions. This type of information processing is completely different from modern computers and AI approaches. Although we have not yet fully understood the mechanisms of the brain, now is the time to create a new type of information science and technology of the brain style.

The present talk gives a general overview of neural networks in the twenty-first century. We need mathematical foundations of neurocomputing and neural systems. Information geometry provides one such method. We discuss the manifold of neural networks and its geometrical structure, where multilayer perceptrons are used as a simplest example. The information geometry of multilayer perceptrons introduces a new efficient learning algorithm, which can be generalized to dynamical computations in neural networks.
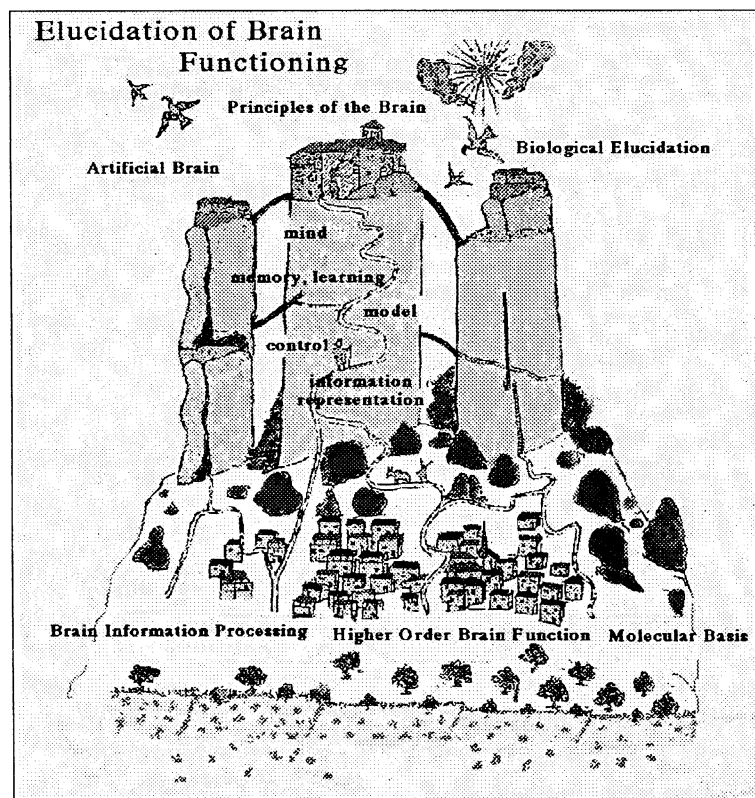
Analysis of Brain and Mathematics of Brain

Toward universal laws of evolving systems

Mark A. Bedau
Reed College

Abstract

A central project in the field of artificial life
has been the search for ways to formulate and
test universal laws of evolution. Such laws, if
they exist, would apply to broad classes of
evolving systems, including both man-made
systems and systems found in nature. The
laws allow us to make meaningful
quantitative comparisons among the
evolutionary activity across evolving systems.
And, most important, the quantitative metrics
in those laws highlight the adaptive aspects of
evolutionary dynamics rather than those
aspects due to chance or necessity. This talk
describes recent progress towards such
universal laws of evolution. The centerpiece
of these laws is the evolutionary activity
statistics recently developed by Bedau and
Packard and co-workers. These statistics
distinguish different dimensions of
evolutionary activity, such as its intensity and
its extent. The centerpiece of these methods is
normalizing the data observed in evolving
systems with appropriate models of neutral
evolution. Application of these methods
yields quantitative comparisons of
evolutionary activity across different evolving
systems, and thus enable us to start making
progress on the search for universal laws of
evolving systems.

Simulation in business and geopolitics

John L. Casti
Santa Fe Institute
Santa Fe, NM, USA

Abstract

The real world of money and war has always
been a difficult place to do experiments---at
least the sort of experiments that are needed
to create a scientific theory of human
interaction. Computer technology now allows
the possibility of just these sorts of
experiments, by enabling us to create large-
scale computer simulations of human conflict
situations.

This talk will present examples of how
computer simulations have been used to
model conflict between companies in the
world's catastrophe insurance industry, as
well as conflict between power alliances in
the world's geopolitical playing field. In both
cases, insights into the way successful
strategies for survival are discussed, as well
as some of the potential pitfalls in using this
kind of analysis as the basis for real-world
decisions.

THE APPLICATION OF LEARNING ALGORITHMS AND UNCERTAIN VARIABLES IN THE KNOWLEDGE-BASED PATTERN RECOGNITION

Z.Bubnicki

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

Abstract

A class of knowledge-based pattern recognition problems with relational and logical knowledge representation is formulated. In the first part, the learning algorithm consisting in using the learning sequence for *step by step* evaluation and updating of the knowledge is described. In the second part, the application of so called uncertain variables and certainty distributions given by an expert is presented for independent recognition problem and for the recognition in a simple decision making system. Simple examples and results of simulations illustrating the general methods are included.

1. Introduction

One of the major areas of artificial intelligence is concerned with learning processes and the knowledge-based pattern recognition. In the papers [1,2,3,4,5] the methods and algorithms of learning for a class of knowledge-based systems described by relational and logical knowledge representations with unknown parameters have been presented. The main idea of learning consists in *step by step* knowledge evaluation and updating, and using the results for the current determination of knowledge-based decisions. It may be considered as an extension of the concept of adaptation via identification for the systems described by traditional mathematical models (see e.g. [6]). For a class of uncertain systems (including knowledge-based systems) with unknown parameters an application of so called uncertain variables and certainty distributions has been proposed and developed [2,7,8].
The purpose of this paper is to show how the methods of learning and the concept of uncertain variables may be applied to a class of the knowledge-based pattern recognition (see [9]) with unknown parameters in the knowledge representation concerning the vector of features

of the object to be recognized and the indexes of classes to which the object may belong. The specification of the pattern recognition problems requires an adaptation and modification of the methods elaborated for the general knowledge-based systems. On the other hand, taking into account numerous practical applications (e.g. computer aided medical diagnosis or the recognition of current situations in intelligent control systems) in which knowledge representations given by an expert contain unknown parameters, we may say that the problems of decision making based on the knowledge representation with unknown parameters are specially important in the area of classification and pattern recognition.
In section 2, the knowledge-based pattern recognition problem under consideration is formulated. The algorithms of learning in two versions are described in section 3. An important special case and a simple example with the results of simulations are presented in section 4. In section 5 we consider the application of the uncertain variables to the pattern recognition as an independent problem and to the recognition in a simple decision making system.

2. Knowledge based pattern recognition

Denote by $x \in X$ a vector of features of the object to be recognized (classified) and by j the index of a class to which the object belongs, $j \in J = \{1, 2, ..., M\}$. A **relational knowledge representation** has the form of a relation $R(x, j; c) \subset X \times J$ where $c \in C$ is a vector of parameters. The relation R is reduced to the sequence of sets

$$D_x(j; c) = \{x \in X: (x, j) \in R(x, j; c)\}, j = 1, 2, ..., M. \quad (1)$$

In the typical cases the sets D_x are determined by sets of inequalities and /or equalities concerning the components of x. The relation R may be presented in a specific form as a set of facts given by an expert (**logical knowledge**

representation), i.e. logical formulas $F_i(\alpha_x, \alpha_w, \alpha_j)$,

$i \in \overline{1,s}$, composed with simple formulas from the sequences α_x, α_w, α_j where α_x is the sequence of the simple formulas (simple properties) concerning x, α_j is the sequence of the formulas: " the object belongs to class j " ($j = 1,...,M$) and α_w is the sequence of the simple formulas concerning x, j and some additional variables [9].

The knowledge based pattern recognition problem may be formulated as follows: given R and D_x ($x \in D_x$ is the result of observations), find the smallest set $D_j \subset J$ such that the implication $x \in D_x \rightarrow j \in D_j$ is satisfied. The set D_j is the set of all possible classes determined by R and D_x :

$$D_j(c) = \{ j \in J : \bigvee_{x \in D_x} x \in D_x(j; c)\}. \qquad (2)$$

For the logical knowledge representation D_x is presented by a logical formula $F_x(\alpha_x)$, i.e. an observed property concerning x, and the pattern recognition problem is formulated as follows: given F_i, $i \in \overline{1,s}$ and F_x, find the smallest set D_j such that the implication $F_x \rightarrow j \in D_j$ is satisfied. The result may be obtained by using a unified algorithm based on **logic-algebraic method** [10, 11].

3. Learning algorithms

For the further consideration we assume that $D_x(j; c)$ are continuous and closed domains in X. Assume now that the vector parameter c has the value $c = \bar{c}$ and $\bar{c}$ is unknown. If we have the learning sequence

$$(x_1, j_1), (x_2, j_2), ..., (x_n, j_n), \bigwedge_i (x_i, j_i) \in R(x, j; \bar{c})$$

then we can propose a current *step by step* estimation of $\bar{c}$. On each step one should prove if the current pair in the learning sequence "belongs" to the knowledge representation determined to this step (**knowledge validation**) and if not, one should modify the current estimation of parameters in the knowledge representation (**knowledge updating**). We shall present two versions of the estimation. In version 1 the relation R is considered as a whole and in version 2 the sets D_x are considered separately.

Version 1. Let us introduce the set

$$D_c(n) = \{c \in C : \bigwedge_{i \in \overline{1,n}} [x_i \in D_x(j_i; c)]\}. \qquad (3)$$

The boundary $\Delta_c(n)$ of the set $D_c(n)$ is proposed here as the estimation of $\bar{c}$. Assume that the pairs (x_i, j_i) in the learning sequence occur randomly with probabilities p_j for j and probability densities $f_j(x)$ for x.

Theorem 1: *If $p_j > 0$ and $f_j(x) > 0$ for every (x, j) $\in R$ and R ($x, j; c$) $\neq R$ ($x, j; \bar{c}$) for every $c \neq \bar{c}$ then $\Delta_c(n)$ converges to $\{\bar{c}\}$ with probability 1.*

The proof is analogous to that presented in [5]. The determination of $\Delta_c(n)$ may be presented in the form of the following **recursive algorithm** for $n > 1$:

Knowledge validation
One should prove if

$$\bigwedge_{c \in D_c(n-1)} [x_n \in D_x(j_n; c)]. \qquad (4)$$

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

Knowledge updating

$$D_c(n) = \{c \in D_c(n-1) : x_n \in D_x(j_n; c)\}. \qquad (5)$$

For $n = 1$

$$D_c(1) = \{c \in C : x_1 \in D_x(j_1; c)\}.$$

On each step we can determine the current set $D_j(c_n)$ by choosing randomly c_n from $\Delta_c(n)$ with a fixed probability distribution for $\Delta_c(n)$. The execution of the algorithm requires using the generator G of random numbers for the random choice of c_n. The block scheme of the learning recognition system is presented in Fig.1.
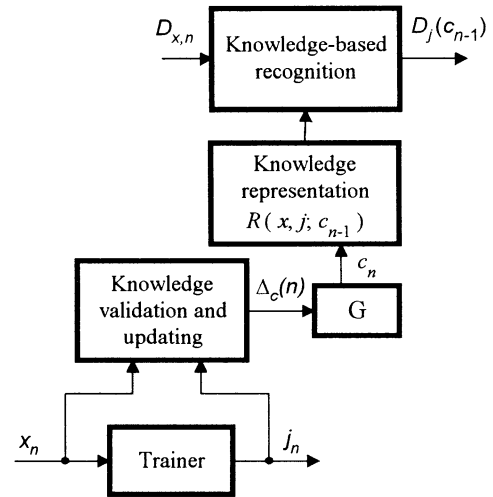
Fig.1

Version 2. Assume that we have different vectors $c_j \in C_j$ in the different classes, i.e. $c = (c_1, c_2, ..., c_M)$ and for each class the set $D_x(j; c_j)$ depends on c_j. Now the estimations $\Delta_{c_j}(n)$ of $\bar{c}_j$ may be determined separately for each class. For the class j, denote by $\bar{x}_{ji}$ the subsequence of x_i for which $j_i = j$, by $\hat{x}_{ji}$ the subsequence for which $j_i \neq j$, and introduce the sets corresponding to (3):

$$\overline{D}_{c_j}(n) = \{c_j \in C_j : \text{ each } \bar{x}_{ji} \in D_x(j; c_j)\}, \qquad (6)$$

$$\hat{D}_{c_j}(n) = \{c_j \in C_j : \text{ each } \hat{x}_{ji} \in X - D_x(j; c_j)\}. \qquad (7)$$

The set

$$\overline{D}_{cj}(n) \cap \hat{D}_{cj}(n) \overset{\Delta}{=} \Delta_{cj}(n) \qquad (8)$$

is proposed here as the estimation of $\overline{c}_j$. The convergence condition and the recursive algorithms for the determination of c_{jn} are analogous to those in version 1. It is easy to note that the presented approach may be extended for general case considered in version 1 where the same parameters c may by used in the description of $D_x(j)$ for the different j. In this case we determine the estimations of $\overline{c}$ in the form of $\Delta_{cj}(n)$ separately for each class using (6), (7) and (8) with c in the place of c_j. Consequently

$$\Delta_c(n) = \bigcap_{j=1}^{M} \Delta_{cj}(n).$$

It is easy to see that version 2 gives better estimation than version 1, i.e. the set $\Delta_c(n)$ obtained in version 2 is smaller than in version 1.

4. Special case and example

Let $M = 2$, $D_x(1; c)$ be a hyperball $x^T x \le c^2$ and $D_x(2; c) = X - D_x(1; c)$. Then using version 2 we obtain

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

where

$$c_{\min,n}^2 = \max_i \overline{x}_i^T \overline{x}_i , \quad c_{\max,n}^2 = \min_i \hat{x}_i^T \hat{x}_i .$$

This result may be extended for $M \ge 2$ where $D_x(j; c)$ is determined by hyperspheres

$$\underline{c}_j^2 \le x^T x < \overline{c}_j^2, \quad c = (\underline{c}_j, \overline{c}_j), \quad \underline{c}_1 = 0, \quad \underline{c}_{j+1} = \overline{c}_j .$$

Consider now a simple example with $M = 2$. According to the general formulations in section 3, the **algorithm of the recognition with learning** is the following:
Introduce (x_n, j_n) from the learning sequence, $n > 0$.
If $j_n = 1$

Knowledge validation for $x_n = \overline{x}_n$
Prove if

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

i.e.

$$(x_n^{(1)})^2 + (x_n^{(2)})^2 \le c_{\min, n-1}^2 .$$

If yes then $\overline{D}_c(n) = \overline{D}_c(n-1)$, i.e. $c_{\min,n} = c_{\min,n-1}$. If not, determine new $\overline{D}_c(n)$.

Knowledge updating for $x_n = \overline{x}_n$

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

i.e.

$$c_{\min, n}^2 = (x_n^{(1)})^2 + (x_n^{(2)})^2 .$$

Put $\hat{D}_c(n) = \hat{D}_c(n-1)$, i.e. $c_{\max,n} = c_{\max,n-1}$.
If $j_n = 2$

Knowledge validation for $x_n = \hat{x}_n$
Prove if

$$\bigwedge_{c \in \hat{D}_c(n-1)} [x_n \in D_x(2; c)]$$

i.e.

$$(x_n^{(1)})^2 + (x_n^{(2)})^2 > c_{\max, n-1}^2 .$$

If yes then $\hat{D}_c(n) = \hat{D}_c(n-1)$, i.e. $c_{\max,n} = c_{\max,n-1}$. If not, determine new $\hat{D}_c(n)$.

Knowledge updating for $x_n = \hat{x}_n$

$$\hat{D}_c(n) = \{c \in \hat{D}_c(n-1) : x_n \in D_x(2; c)\}$$

i.e.

$$c_{\max, n}^2 = (x_n^{(1)})^2 + (x_n^{(2)})^2 .$$

Put $\overline{D}_c(n) = \overline{D}_c(n-1)$, i.e. $c_{\min,n} = c_{\min,n-1}$.
Choose randomly c_n from
$$\Delta_c(n) = \overline{D}_c(n) \cap \hat{D}_c(n) = [c_{\min, n}, c_{\max, n}).$$
Introduce the result of observations $D_{x,n}$.
Determine $D_j(c_n)$:

$$D_j(c_n) = \begin{cases} \{1\} & \text{if } D_{x,n} \subseteq D_x(1; c_n) \\ \{2\} & \text{if } D_{x,n} \subseteq D_x(2; c_n) \\ \{1,2\} & \text{otherwise}. \end{cases}$$
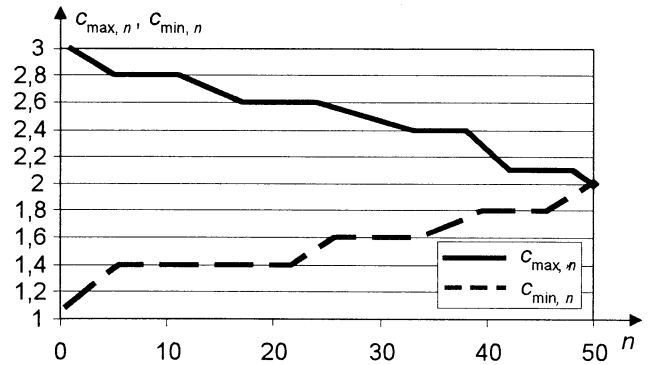
Fig. 2

For this case the learning and pattern recognition process has been simulated for the different probability distributions for x and the different numerical values of x_1 and c. The results are presented in Fig.2 for $\overline{c} = 2$. The values x were chosen randomly with the probability density $f(x)$ constant in the circle with the radius $c = 5$

and equal to 0 outside this circle. The value c_n was chosen randomly from $[c_{min,n}, c_{max,n})$ with rectangular probability distribution. The simulations for different cases showed the significant influence of $f_j(x)$ and p_j on the convergence of $\Delta_c(n)$ to $\bar{c}$. For $n > 50$ $c_{min,n} \approx c_{max,n} \approx 2$ and the recognition based on $D_x(j; c_n)$ is practically correct if $D_x = \{x\}$ (a singleton).

5. Application of uncertain variables

Let us assume that $\bar{c}$ is an uncertain variable described by a certainty distribution $h(c) = v(\bar{c} \approx c)$ where $v(\bar{c} \approx c)$ is the certainty index of the soft property " $\bar{c}$ is approximately equal to c". To formulate the recognition problem and obtain the result we may apply the general approach to uncertain systems analysis with the application of uncertain variables [2,7,8]. Consider the case with $D_x(j; c_j)$ in version 2 from section 3 and suppose that for each j and each $m \neq j$

$$D_x(j; \bar{c}_j) \cap D_x(m; \bar{c}_m) = \varnothing \text{ (empty set).} \quad (9)$$

The recognition problem may be formulated as follows: given $h_j(c_j)$ for $j = 1, 2, ..., M$ and D_x, find $j = j^*$ for which the certainty index of the property " the object belongs to class j " is the greatest. Denote by $\bar{j}$ the correct classification (i.e. the object belongs to class $\bar{j}$, but $\bar{j}$ is unknown). Then

$$v(\bar{j} = j) \overset{\Delta}{=} v_j(D_x) = v[\bigwedge_{x \in D_x} x \overset{\sim}{\in} D_x(j; \bar{c}_j)] = v[\bar{c} \overset{\sim}{\in} D_c(j, D_x)]$$

where $\qquad\qquad x \in D_x \qquad\qquad\qquad\qquad (10)$

$$D_c(j, D_x) = \{c_j \in C_j : D_x \subseteq D_x(j; c_j)\} \quad (11)$$

and $a \overset{\sim}{\in} A$ is a soft property "a approximately belongs to A". Then

$$v_j(D_x) = \max_{c_j \in D_c(j; D_x)} h_j(c_j) \quad (12)$$

and

$$j^* = \arg\max_j v_j(D_x).$$
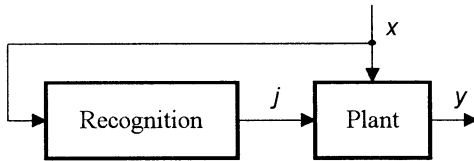
Fig. 3

If the assumption (9) is not satisfied then $v_j(D_x)$ obtained by (10) and (11) denotes the certainty factor of the property "it is possible that $\bar{j} = j$" or "it is possible that j belongs to the set (2) with $c = \bar{c}$ ". Hence, j^* is the index of the class for which the certainty factor of this property is the

greatest. The determination of the values $v_j(D_x)$ may be useful for the evaluation of the dispersion.

An interesting problem arises when j is a decision for a plant with an output vector $y \in Y$ (Fig.3), described by a relational knowledge representation $R(x, j, y; \bar{c})$ with an unknown vector parameter $\bar{c}$. The decision problem, and consequently the recognition problem may be now formulated as follows: given the result of observations x, the set $D_y \subset Y$ (the property $y \in D_y$ is a user's requirement) and $h(c)$, find the best decision j^* maximizing the certainty index $v(\bar{y} \overset{\sim}{\in} D_y)$. Using the general approach to decision making for uncertain systems presented in [7,8] we determine

$$D_y(x, j; c) = \{y \in Y : (x, j, y) \in R(x, j, y; c)\} \quad (13)$$

and

$$D_j(x; c) = \{j \in J : D_y(x, j; c) \subseteq D_y\}. \quad (14)$$

Then

$$v(\bar{y} \overset{\sim}{\in} D_y) \overset{\Delta}{=} v_j(x) = v[j \in D_j(x; c)] = v[\bar{c} \overset{\sim}{\in} D_c(x, j)]$$

where

$$D_c(x, j) = \{c \in C : j \in D_j(x; c)\}. \quad (15)$$

Finally

$$v_j(x) = \max_{c \in D_c(x, j)} h(c) \quad (16)$$

and

$$j^*(x) = \arg\max_j v_j(x).$$

Thus we obtain the recognition algorithm $j^*(x)$ which for the given x determines the value j for which the requirement $y \in D_y$ is satisfied with the greatest certainty index.

Example

Let $j \in \{1, 2, 3\}$; $x, y \in R^1$ (one-dimensional input and output). The relation $R(x, j, y; c)$ is determined by the following description:

for $j = 1$ $\qquad x + c \leq y < x + 2c,$

for $j = 2$ $\qquad x + 2c \leq y < y + 3c,$

for $j = 3$ $\qquad y + 3c \leq y < y + 4c.$
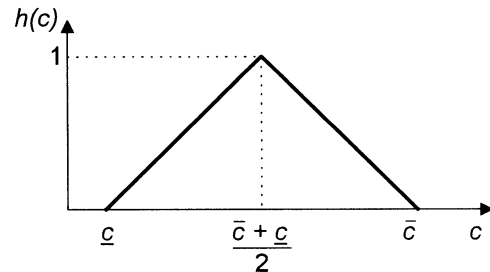
Fig. 4

Assume $x > 0$ and $D_y = [\underline{y}, \bar{y}]$. Then, according to (13), (14) and (15) the sets $D_c(x, j)$ are determined as follows:

for $j = 1$ $\quad \underline{y} - x \le c \le \dfrac{\bar{y} - x}{2}$,

for $j = 2$ $\quad \dfrac{\underline{y} - x}{2} \le c \le \dfrac{\bar{y} - x}{3}$,

for $j = 3$ $\quad \dfrac{\underline{y} - x}{3} \le c \le \dfrac{\bar{y} - x}{4}$.

Suppose that $h(c)$ has the triangular form presented in Fig.4. For numerical data $\bar{y} = 8$, $\underline{y} = 6$, $x = 2$, $\underline{c} = 1$, $\bar{c} = 13$ we obtain:

for $j = 1$ $\quad \underset{c}{\arg\max}\, h(c) = 6,$ $\quad v_1 = h(6) = \dfrac{5}{6},$

for $j = 2$ $\quad \underset{c}{\arg\max}\, h(c) = 4,$ $\quad v_2 = h(4) = \dfrac{1}{2},$

for $j = 3$ $\quad \underset{c}{\arg\max}\, h(c) = 3,$ $\quad v_3 = h(3) = \dfrac{1}{3}$

and we accept $j^* = 1$ with the uncertainty index $\dfrac{5}{6}$. For $\underline{c} = 1$ and $\bar{c} = 6$ we obtain:

for $j = 1$ $\quad v_1 = h(4) = \dfrac{4}{5},$

for $j = 2$ $\quad v_2 = h(3.5) = 1,$

for $j = 3$ $\quad v_3 = h(3) = \dfrac{4}{5}$

and $j^* = 2$.

6. Final remarks

The simulations and numerical examples for the different cases and data show the significant influence of the parameters in the learning algorithms and in the certainty distributions on the quality of the recognition. It should be taken into account in the design and implementation of expert systems based on the presented method. The methods and algorithms presented shortly in the paper may be extended for more complicated recognition systems (in particular, for two-level pattern recognition and learning control systems containing the recognizer). Interesting and important problems arise when we try to combine two approaches presented in the paper: using *a priori* information in the form of certainty distributions given by an expert and current information in the learning process. Such an approach has been mentioned in [4] and is now the subject of further considerations.

The learning algorithms and the application of uncertain variables for the knowledge-based pattern recognition have been used in the modification of expert system CLASS-LOG oriented for the recognition and classification problems.

References

[1] Bubnicki Z (1998), Learning processes and logic-algebraic method in knowledge-based control systems. In: S.G. Tzafestas and G. Schmidt (eds), Progress in System and Robot Analysis and Control Design. Lecture Notes in Control and Information Sciences. Springer Verlag, London, pp. 183-194

[2] Bubnicki Z (1999), Uncertain variables and learning algorithms in knowledge-based control systems. Artificial Life and Robotics, Vol.3 (in press)

[3] Bubnicki Z (1999), Learning algorithms in a class of knowledge-based systems. In: M. Sugisaka (ed), Proceedings of the International Symposium on Artificial Life an Robotics (AROB 4th), Beppu, Oita, Japan, 1999, pp. OP-34 - OP-39

[4] Bubnicki Z (1999), Methods and algorithms in learning control systems. In: Proceedings of the 13th International Conference on Systems Engineering, Las Vegas, USA, 1999, pp.EE-19 - EE-24

[5] Bubnicki Z (1999), Learning control systems with relational plants. In: Proceedings of European Control Conference, Karlsruhe, Germany, 1999

[6] Bubnicki Z (1980), Identification of control plants. Elsevier, Oxford – Amsterdam – New York

[7] Bubnicki Z (1998), Uncertain variables and logic-algebraic method in knowledge-based systems. In: M.H. Hamza (ed), Intelligent Systems and Control. Proceedings of IASTED International Conference, Halifax, Canada, 1998, pp. 135-139

[8] Bubnicki Z (1998), Uncertain logics, variables and systems. In: L. Guangquan (ed), Proceedings of the 3rd Workshop of the International Institute for General Systems Studies, Beidaihe, Qinhuangdao, China, 1998, pp. 7-14

[9] Bubnicki Z (1993), Knowledge-based approach as a generalization of pattern recognition problems and methods. Systems Science 2:5-21

[10] Bubnicki Z (1997), Logic-algebraic method for a class of knowledge based systems. In: F. Pichler and R. Moreno-Diaz (eds), Computer Aided Systems Theory. Lecture notes in Computer Science. Springer Verlag, Berlin, pp. 420-428

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

Compression and Adaptation

Charles E. Taylor and Tracy Teal
Department of Biology, 2203 Life Sciences
University of California, Los Angeles
taylor@biology.ucla.edu

Abstract

What permits some systems to evolve and adapt more effectively than others? Gell-Mann has stressed the importance of ``compression" for adaptive complex systems. Information about the environment is not simply recorded as a look-up table, but is rather compressed in a *theory* or *schema*. Several conjectures are proposed:

(I)   compression aids in generalization;

(II)   compression occurs more easily in a ``smooth", as opposed to a ``rugged", string space; and

(III)   constraints from compression make it likely that natural languages evolve towards smooth string spaces.

We have been examining the role of such compression for learning and evolution of formal languages by artificial agents. Our system does seem to conform generally to these expectations, but the tradeoffs between compression and the errors that sometimes accompany it need careful consideration.

A Review to The International Symposium on Artificial Life and Robotics (AROB)

Masanori Sugisaka
(Dept. of Electronic and Electrical
Engineering, Oita University,
Oita, Japan, 811)

msugi@cc.oita-u.ac.jp

Y.G. Zhang
(Institute of Systems Science, Academia
Sinica, Beijing, China, 100080
yzhang@iss01.iss.ac.cn
& ATR, Human Information Processing
Research Laboratories, Kyoto, Japan. 619)

Abstract

In this paper we reviewed the whole process of growth with AROB conference, and we have sorted the various contributions in AROB, that including digital life, artificial brain, artificial society and evolutionary robots, etc., Analyzed the relations between Artificial Life, complexity and intelligent robots. Finally, gave the expected future of AROB.

Key words: Artificial Life; digital life; primitive artificial brain, evolutionary robots; Santa Fe Institute; ATR

This year is the birth of a new millennium and our AROB'00 is held in the January of 2000. That means earliest step into the new millennium. In addition, this AROB'00 is the fifth anniversary of International Symposium on Artificial Life and Robotics. All those who take care of and pay more attentions to its growth are very glad to celebrate this wonderful academic festival, because this baby is already five years old now! AROB is really an international symposium. The presenters on AROB are from about 15 countries. The total submissions in each AROB is now more than 200 papers. Most of professors and engineers in the world who work on the fields of Artificial Life and Robotics know it. AROB have received huge support from Japanese government and industrial groups. They hope AROB contribute over the whole world more and more. It will be very meaningful and necessary to recall the progress in this five years and summary it during this AROB'00.

I. The birth of AROB

As we know, the original source place of Artificial Life is in Santa Fe Institute in US., and the first symposium on Artificial life was held in 1987 in Los Alamos of US.. Dr. Chris Langton held the symposium as chairman. Since then the "Artificial Life" was born and a "digital Life" can be shown on a computer screen. This methodology

of thinking has affected many scientists including whom in computer science, robotic engineering, system science, biology, etc.

In January of 1994 Professor M. Sugisaka had invited Professor J. Casti of the Santa Fe Institute to his laboratory in order to perform a collaborative research "**Advanced neural control of intelligent mobile vehicle**" supported by Japan Society for the Promotion of Science (JSPS) for three weeks. His invitation was recommended by famous Prof. Emeritus of Kyoto University S. Ueno, who is a distinguished astrophysicist. Professor M. Sugisaka warmly welcomed Prof. J. Casti's visit in Japan. Related this visit Professor M. Sugisaka discussed collaborative researches with Santa Fe Institute in the future with him. This led to more exchange between Japan and Santa Fe Institute. During this period Professor M. Sugisaka made contact with Professor T. Ray either. Then he was invited to visit Professor Sugisaka's laboratory. In addition, during their visits and after there were other visiting professors to perform collaborative researchs on **Artificial Life** with Professor M. Sugisaka. They have a common thinking that **Artificial Life** is a quite new field and it has a very strong vitality in the new century. In the period a Department on Evolution Technology has been established in ATR Human Information Processing

Laboratory, Kyoto, Japan. "Artificial life" has already been an important project there.

Professor Sugisaka has thought a new idea that is there some certain relations between Artificial life and Robotics? How can a robot have "life" and a clever "brain"? There is very advanced robotic technology in Japan, and many professors and engineers who engage the research on intelligent robots and autonomous robots. They should join this new research field together. Many Japanese professors accepted Prof. Sugisaka's thought. Also, some professors in China and Korea have backed up this idea. Professor Sugisaka has made decision to push the research on **Artificial Life** and **Robotics.** The more important thing is that this innovation proposal has got the huge supports from the Ministry of Education, Science, Sports, and Cluture(Monbusho) of Japanese government and some industrial groups. Then, the International Organization Committee of AROB formed. At last, as all you know, the first AROB'96 held in the February of 1996. AROB was born now.

II. The growth of AROB

Since the first AROB'96 in 1996, it has many progress and development. Two years later, more famous professors joined AROB. They come from UCLA, Los Alamos National Lab. of U.S.A., GMD in Germany, Tsinghua University and Academia Sinica in China, Australia, Switzerland, Canada, Poland, British, Korea, etc. No doubt, AROB is already a real international symposium!

The statistical number below is the evidence.

Table 1 The statistic for AROB

Year	The number of presentation	The number of countries joined
1996	72	6
1997	38	5
1998	186	9
1999	206	9
2000	240	15

In addition to the symposium and its proceedings of AROB, now AROB has already had a formal **Journal** named **"Artificial Life and Robotics"**, and published by Springer, Tokyo. Now it has published in its volume 3.

So far, there are three main conferences on Artificial Life all over the world. They are:

Artificial Life (ALife)---organized by United States, The first three conferences is organized by Santa Fe Institute in US, it has passed six conferences that: Alife I (1987), Alife II (1989), AlifeIII (1992), Alife IV(1994), Alife V(1996), Alife VI (1998). There will be Alife VII in August, 2000.
Publications: Proceedings and Journal "ARTIFICIAL LIFE".

Artificial Life and Robotics (AROB) --- organized by Oita University, Japan. You can find five conferences that: AROB'96(1996)[1], AROB'97(1997)[2], AROB'98[3], AROB'99, AROB'99 (1999)[4] and AROB'00(2000)[5].
Publications: Proceedings and Journal "ARTIFICIAL LIFE AND ROBOTICS".

European Conference on Artificial Life (ECAL) --- organized by Europe, so far five conferences were held in Europe countries. They are that: ECAL'91(1991), ECAL'93 (1993), ECAL'95(1995), ECAL'97(1997) and ECAL'99 (1999).
Publications: Proceedings.

III. The Contributions of AROB

Based on the previous statistics the total of AROBs presentations are 742 papers (including this AROB'00), which covers various directions of Alife. We could sort them from various viewpoint and give comments to each if we want according to the ordinary rules. However, it is not enough to summarize the whole contributions of AROB. The problem is that we face to a quite new field and so far there is not a definition to Artificial Life accepted by most of scientists. So, our task is not simply to classify the papers, but also to express our opinion and viewpoints about what is the Artificial Life and how will it develop? Then we can understand what are the contributions of AROB.

The main contributions of AROB are in two aspects: 1) As mentioned in a call for papers of AROB that the combination of artificial life and robotics is the most important feature of this symposium. This means that artificial and real worlds will encounter and be fused there; 2) Attracted more researches on Artificial Life, specially in Asia area, and made more clear with the research content of Artificial Life. More details are explained below:

A) The original new thought about Artificial Life is from Santa Fe Institute and opened to the world is in 1987. However, its propagation is not so fast and wide as Artificial Neural Network (ANN) and Genetic Algorithm (GA), most of researchers are located and centralized in some universities in US. The earliest research group on Artificial Life in Asia area is in Human Information Processing Research Laboratory, ATR, in Kyoto of Japan. Dr. Shimohara leads this department in which including Tom Ray and Hugo Garis and some others. Many scientists did not know what "Artificial life" is and what the content of research is before 1995. Especially in Asia area, only a few professors heard about this terminology. How to attract more scientists and engineers through their research into this new field in which there is very important new scientific thought and philosophy, and it has bright future in 21st century? AROB has done this job, more and more scientists and engineers through themselves into this new area, especially in Asia area, for example, Japan, China, Korea, Singapore and Hong Kong etc.

B) The research of "**Digital life**" is a very basic one in Artificial Life. So-called "Digital life" means specially that the "life" with medium of computer RAM space and CPU time, the 'body' is a program written by certain computer language. A digital life should be designed as such a computer program that can survive in the environment of silicon memory and it can be copied or reproduced, it might have metabolism function and could be dead, it is evolvable in certain conditions from one-cell to multi-cell till then an organization, and from a simpler into a more complicated in morphology. The complexity become higher and higher as its evolution. Digital life is absolutely not a simulation or modeling to the real cell, but an artificial system in which "life" has similar meaning and digital medium. Digital life is also an experiment to test if a "life" could survive in non-protein environment and medium? Some good digital life system have shown at AROB, for example, the Tierra by Tom Ray who is in ATR, it is very helpful to understand what is artificial life. Some further works are expected that is a theoretical frame to describe the common characteristic of digital life, and what is the **evolvability** for them? and how to evaluate it? This measure may be helpful to the evolution research in Biology.

C) "**Artificial brain**" would be a very attractive and challenged technology in the future, the software experiment has finished and a hardware prototype has being developed in recent years. The design of CAM-Artificial Brain by Hugo Garis in ATR is really based on Artificial Life principle and to simulate real brain structure. It is made by evolution and growing up. We do not know if it will be successful surely or not, but we believe that Artificial Brain designed according to Alife principle must be created in the next century. At least this kind of design will be benefit to the design of new generation of neural computer.

Before finishing the implementation of this Artificial Brain, we can construct a "**Primitive Artificial Brain**" by the present neural computer. This is a very practical design for intelligent robots. It has visual signal processing and decision-making ability to deal with the information from CCD camera, it has a similar process to a real brain and to make robots more clever. Of cause, it is still very simple compared with true brain function. This kind of Primitive Artificial Brain has already implemented in the AROB Lab. of Oita University led by Professor Sugisaka.

D) **Evolutionary Robots** is a connection point between Robotics and Artificial Life. Intelligent robots maybe or maybe not have some artificial life meaning, but evolutionary robots have the artificial life meaning definitely. Its design is based on the evolution principle and different from the mechanical design or usual autonomous system. Evolution robot is autonomous, of cause, and his some function is formed by "evolution" and "emergency". In addition, the evolutionary algorithms can be embedded into the robots easily. Many participants interest the works by Prof. Fukuda, Prof. P. Pfeifer and Dr. Gomi. The design for evolutionary robots will get more new idea from Artificial Life and give more exciting applications.

In recent AROB's the robots soccer match is very attractive, the intelligence in each robot agent is quite different. The winner maybe more clever and trick than the others due to it was "born" in evolution.

Agricultural robot is a quite advanced experiment. It should has more flexibility to adapt to a very worse working circumstantial, it may be more benefit for its design to get some idea from Artificial Life and Evolutionary robot.

E) **Artificial ecology** and **Artificial society**. Some invited speakers introduced these new ideas. Basically, these ideas are to show the evolutionary process of their digital creatures or economy agents, for example, the "Virtual Creatures" by Karl Sims and the "Sugarscape" by J. M. Epstein. They used the virtual reality (VR) technique to deal with visualization of the evolution process. If the research in digital life borrows this idea then it could be very attractive and visualized then previous.

F) **Evolutionary computation** is an adjacent of Artificial Life. The evolutionary computation includes the whole class of mathematical algorithms such as Genetic Programming (GP), Genetic Algorithm (GA), etc, in which there is certain life meaning and evolution operations, for example, reproduction, crossover and mutation. If we want to develop any evolutionary operation in digital style, to use certain mathematical algorithm is necessary. At present, whatever the theoretical research on the algorithms or application research all are needed. The papers on evolutionary computation have a certain percentage of the whole presentations.

We observed some interesting development in the application of evolutionary computation. One is the combination between some evolutionary algorithm and artificial neural network (ANN) to solve the on-line learning problem; the other is to solve the problem of cooperative behavior of autonomous mobile robots. These are all very good applications of evolutionary computation to make robots be more humanlike.

G) The electronic **Arts related to Artificial Life**. These works, in fact, are very attractive then many other research papers due to their fantastic visualized screen image. The creatures move just like a real animal. They are not usual cartoon by drawing something on screen; of cause they may be the future's cartoons surely. It is a real art with the combination of arts, science and engineering. The works by Christa Sommerer in ATR made participants opened their eyes to see what is inside. Her new work named "Life Species" has more life meaning than her previous, T. Ray said **"evolution is life's arts"**, however, she could give us an **"arts with life"**.

H) AROB has held several **invited speaks** and **panel discussion** in its series symposiums. You can find the detailed list from the programs of each AROB symposium. Those speakers showed the participants many interesting ideas and new advance of Alife and related area. Those bring to AROB more colorful and highlight it. The honorable and distinguished speakers include (the list is ordered by the date and time who give speak, but any other reasons):

J. Casti (Santa Fe Institute, USA),
C. Langton (Santa Fe Institute, USA),
T. Musha (Brain Function Lab, Inc., Japan),
T. Ray (ATR, Santa Fe Institute, USA),
H. Miura (University of Tokyo, Japan),
J. Epstein (The Brookings Institution and Santa Fe Institute),
T. Ifukube (Hokkaido University, Japan),
C. Sommerer (ATR Advanced Telecommunication Research, Japan),
M. Otsuka (Nippon Zoki Pharma.Co.Ltd., Japan),
M. Ito (The Institute of Physical and Chemical Research, Japan),
M. Sugisaka (Oita university, Japan)
G. Matsumoto (The Institute of Physical and Chemical Research, Japan),
C. Barrett (Los Alamos National Labs., USA),
R. Palmer (Santa Fe Institute),
J. Doyle (California Institute of Technology, USA),
T. Fukuda (Nagoya University, Japan),
Y. Fujita (The Japan Academy, Japan),
G. James (Coventry University, UK),
J. J. Lee (Korea Advanced Institute of Science and Technology, Korea),
Y. G. Zhang (Academia Sinica, China),
C. Taylor (University of California-Los Angels, USA),
H. Kimura (University of Tokyo, Japan),
T. Oshima (Tokyo University of Pharmacy and Life Science, Japan),
S. Rasmussen (Los Alamos National Laboratory, USA),
R. Belew (University of California-San Diego, USA),
W. Wells (University of Nevada-Las Vegas, USA),
Z. Bubnicki (Wroclow University of Technology, Poland),
F. Harashima (Tokyo Metropolitan Institute of Technology, Japan),

S. Amari (The Institute of Physical and Chemical Research, Japan), M. Bedau (Reed College, USA)

In addition, AROB'96 had a panel discussion named "Coexistences between human being and robots including artificial life"; AROB'98 had a panel discussion named "The status of research into artificial life and related robotics". These discussions were very benefit to understand the relations between artificial life and robotics. Also, the discussion concerns with the relation between artificial life and complexity. All those topics are very new and important for the development of science and technology in 21st century.

IV. Artificial Life and Complexity

Complexity is a quite new topic for many persons in spite of the research has been done for more than ten years in Santa Fe Institute of US, but has not a unified definition recognized by all scientists for it till now. This is a quite difficult problem for the new science field developed fast in the recent years. There is certain relation between the Artificial Life and Complexity. In fact, the both research fields are coexistence in Santa Fe Institute. Alife is a special form to study the complexity. The objects studied by Alife are those complex systems which have "life" characteristics. Real life system is very complicated in structures, functions, information processing and control, especially evolutionary process. The artificial life-systems have certain complexity, but not like the real life systems so far. We could see the increasing of complexity in the evolutionary process of digital life creatures. Also, we could see the increasing of complexity in evolutionary robots and related primitive artificial brain. Especially, the proceedings in AROB'98 you might find a bold headline in the proceedings that "**special contribution to complexity**", that means AROB paid a special attention to the research on complexity and the relation between them.

V. The future of AROB

This is a question concerned with by many persons. Of cause, we will continue encourage the researches on digital life, evolutionary robots, artificial brain and other related works. No doubt we will stretch AROB into more related research fields, for example, multiple agents, neuroscience, hardware evolution, etc.

To attract more persons to attend AROB, we suggest that to make a turn to hold AROB in Asia area, this is more expected in the future. AROB is forward even more international. We hope AROB could hold in China, Korea, etc, in the near future.

Also, to establish an academic society of artificial life and robotics were strongly expected in the near future. (The Alife VII will discuss the same question about establishing an International academic society of Artificial life). This society should back up and organize each AROB symposiums and manage the journal of AROB and its publications. It should make more academic exchanges with other societies related Alife and Robotics, especially with Artificial Life conference in the US and ECAL in Europe. All in one words, AROB has a very bright future.

VI. Acknowledgements

REFERENCES
[1] Proceedings of the First International Symposium on Artificial Life and Robotics (AROB'96), edited by M. Sugisaka, 1996
[2] Proceedings of the Second International Symposium on Artificial Life and Robotics (AROB'97), edited by S. Fujimura and M. Sugisaka, 1997
[3] Proceedings of the Third International Symposium on Artificial Life and Robotics (AROB'98), edited by M. Sugisaka, 1998
[4] Proceedings of the Fourth International Symposium on Artificial Life and Robotics (AROB'99), edited by M. Sugisaka and H. Tanaka, 1999
[5] Proceedings of the Fifth International Symposium on Artificial Life and Robotics (AROB'00), edited by M. Sugisaka and H. Tanaka, 2000
[6] Journal of **Artificial Life and Robotics,** Springer, Tokyo, Vol. 1-4. 1997, 1998, 1999

Measures of Evolvability in Tierra

Thomas S. Ray and Chenmei Xu

ATR Human Information Processing Research Laboratories
2-2 Hikaridai, Seika-cho Soraku-gun, Kyoto 619-02 Japan
ray@hip.atr.co.jp cxu@hip.atr.co.jp
http://www.hip.atr.co.jp/~ray/

Abstract

A variety of indices are tested for their ability to discriminate periods of evolution from periods of stasis in Tierra. Measures which recognize the emergence of new adaptations are favored. To remove artifacts, the shadow method of Bedau is used. A promising measure tracks the positive derivative of the maximum proportion.

Introduction

In building artificial evolving systems, we find that some of these systems exhibit rich evolution, and others don't. When making modifications to systems that do exhibit rich evolution, we find that subtle changes may have a large impact on various properties of the evolutionary process.

Ray (1994) compared the patterns of evolution in four different (but very similar) machine languages. It was found that two of the four showed a much greater magnitude of evolution than the others (measured as optimization through size decrease). Also, the two that showed relatively little evolution, showed a pattern of strict gradualism of evolution, whereas the other two showed abrupt jumps in evolution (punctuations). Of the two showing punctuations, one demonstrated gradual evolution between the punctuations, while the other showed strict stasis between the punctuations.

It is evident that many aspects of the evolutionary process depend on the structure of the underlying genetic language. Yet, there exists no body of theory to guide in the design of enhanced evolvability in evolving systems. This now presents a serious problem for the many engineers who work with evolution as a tool in design or optimization.

In every case, the engineer creates a genetic system to describe the solution space, and then evolves within that language. Some of these languages will be highly evolvable, while others will not. There is no theory to guide the design of these languages to enhance their evolvability. This represents a hole in our evolutionary theory which was not evident before the advent of synthetic evolution.

The work presented here is an effort to address this problem by developing methods to quantify evolvability. If a quantitative measure of evolvability is available, then it should be possible to systematically alter features of evolving systems in order to gather empirical evidence on how those features impact evolvability. In this way we can build a body of data as a foundation for a theory of evolvability.

Methods

Discriminating Evolution from Stasis

The current work explores the development of a quantitative measure of evolvability in the Tierra system, Ray (1991). When Tierra is run under standard conditions, there is a strong tendency for the size of the replicators to decrease by evolutionary optimization, from the initial size of eighty bytes, to as few as twenty-two bytes. Different runs reach different final optima, such as twenty-two, twenty-seven, or thirty bytes. However, in every case, a period of optimization is followed by a period of final stasis (Figure 1).

Figure 1 - Size 990629T

Time in Millions
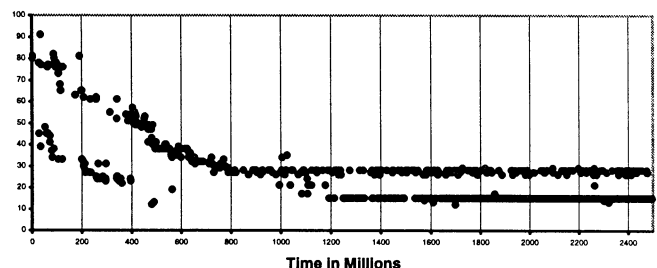
We suggest that a measure of evolvability should at least be able to discriminate the period of evolutionary optimization from the period of final stasis. Thus on 29-6-99, Tierra was run for a period of about twice the time needed to reach stasis (Figure 1).

While the figure shows that a single phenotypic character, size, reaches a stable state, it does not necessarily follow that evolution has ceased in the later part of the run. A closer look is needed to support this claim.

In the run of 29-6-99, genotype 27aal appeared at t = 800 and dominated until the end of the run at t = 2500. At t = 1200 a 15 byte parasite entered into an ecologically and evolutionarily stable equilibrium with the 27 byte host. The 27 byte replicator includes three loci at which two instructions are acceptable (eight neutral mutants). In addition there are viable variants including one-byte insertions or deletions. The 15 byte parasite includes one locus at which two instructions are acceptable and one locus at which any (of thirty-two) instructions are acceptable (sixty-four neutral mutants). From time of the appearance of the fifteen byte parasite through the end of the run, the system was dominated by the eight neutral variants of the host, and the sixty-four neutral variants of the parasite.

Measures of Evolutionary Activity

The data saved from the run was tested with a variety of measures, to find those that can effectively discriminate evolution from stasis.

The ideal measure of evolvability would have a zero value when there is no "evolutionary activity", and a positive value when there is activity. It would have larger values when the activity is greater. Adaptation is the evolutionary event of greatest interest. Adaptation is an evolutionary discovery that provides a relative fitness advantage. Relative fitness advantage is expressed as an increase in frequency. We can measure evolutionary activity by measuring increases in frequency. While we have tried to develop measures with these ideal properties in mind, we have readily tested measures without these properties, in order to prevent our preconceptions from blinding us to useful measures.

We have examined the properties of numerous measures, of which four are presented here: 1) Number – a count of the total number of organisms of all genotyes in the system. 2) Maximum Integral – the largest integral of any genotype population. 3) Maximum Proportion – the proportion of the most common genotype. 4) Positive Derivative of Maximum Proportion – the positive derivative of the third index.

All measures use a threshold of two. This means that any genotype class with only a single individual is completely ignored. These genotypes are not computed in the statistics. This is because the numbers of such genotypes is driven by the mutation rate, and they are mostly inviable mutants.

Shadow Normalization

Measure can be misleading due to artifacts. The fact that the average size of the replicators decreases during evolution but not during stasis can affect some measures, completely apart from the adaptive nature of the process. We need a reliable and consistent way of removing these artifacts. We have borrowed the "shadow" method of Bedau and Brown (1999). The shadow is a null model: everything is random.

The Tierra shadow is directly driven by an actual run of Tierra, it shadows the run. When there is a birth (or death) in Tierra, there is a birth (or death) in the shadow. If a birth in Tierra generates a new size (and therefore also a new genotype), a new size (and genotype) is created in the shadow. If a birth in Tierra generates a new genotype of an existing size, and individual in the shadow is chosen at random, and a new genotype of its size is created. If a birth in Tierra is an existing genotype, an individual in the shadow is chosen at random, and a new individual of that genotype (and size) is born.

The sizes and genotypes in the shadow are arbitrary labels, they have no properties. There is no selection, and no evolution in the shadow. The shadow has identical statistical properties to Tierra, except that there is no selection, no adaptation, no evolution. Any statistic computed on Tierra can be computed on the shadow. Tierra statistics can be normalized by dividing by the shadow statistic (or by subtracting the shadow value). Because the shadow is generated at random, a sample of shadows are used, and the Tierra statistic is normalized by the average of the sample of shadows. In this study, a sample of eight shadows was used in normalization. Shadow normalization leaves only the effects due to evolution.

Results

We will present the results of analysis of each of four potential measures of evolvability.

Number

The number of replicators in the population would appear to be a good index for discriminating evolution from stasis, because the value rises during the period of evolution, and remains constant during the period of stasis (Figure 2, lower curve). However, this behavior is an artifact of optimization: as the replicators decrease in size, more individuals can fit in the constant size environment. This provides an opportunity to illustrate the use of shadow normalization.

Figure 2 shows the graph of Number (lower curve) plotted together with Number generated from the average of eight shadows (upper curve). Because there is a birth in the shadow for each birth in Tierra, and a death in the

Figure 2 - Number
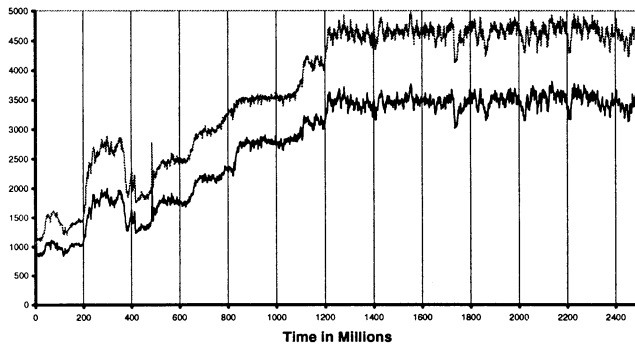
shadow for each death in Tierra, the total number of individuals is always identical in Tierra and its shadow. However, all statistics are computed based on the threshold of two individuals per genotype. Thus, genotypes that contain only a single individual are not counted. The values of Number are lower in Tierra than in the shadow because a greater proportion of genotypes in Tierra have only a single individual and are not counted.

Figure 3 - Normalized Number
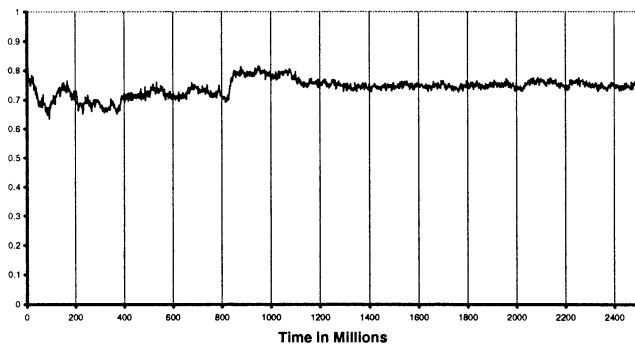
When Number is normalized by dividing the actual value by the average shadow, the curve becomes virtually flat, and it is clear that Number does not discriminate the period of evolution from the period of stasis (Figure 3). Shadow normalization can be applied to any measure of evolvability.

Figure 4 - Normalized Maximum Integral
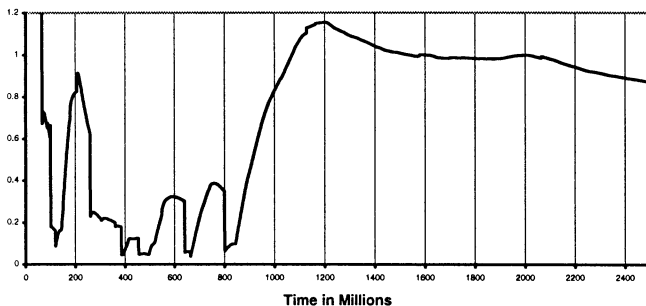
Maximum Integral

The maximum integral is calculated by integrating the population of each genotype (over the threshold) over time. At each time interval, the value of the maximum integral is plotted (Figure 4).

Maximum Proportion

At each time interval, we plot the frequency of the most abundant genotype (Figure 5).

Figure - 5 Normalized Maximum Proportion
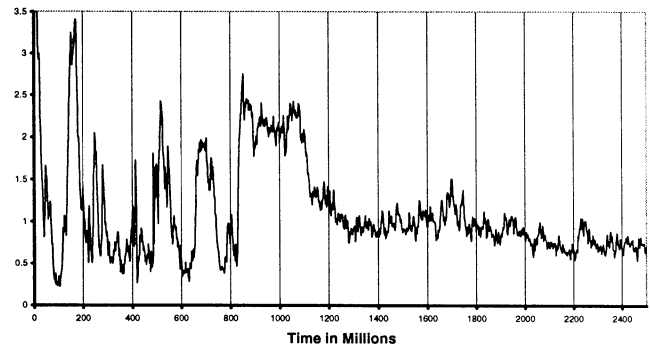
Positive Derivative of Maximum Proportion

At each time interval, we calculate the derivative of the Maximum Proportion. If the derivative is positive, we plot the value, if the derivative is negative, we plot zero. Due to the zero values, this index must be normalized by subtraction rather than division (Figure 6).

Figure 6 - Normalized Derivative of Maximum Proportion
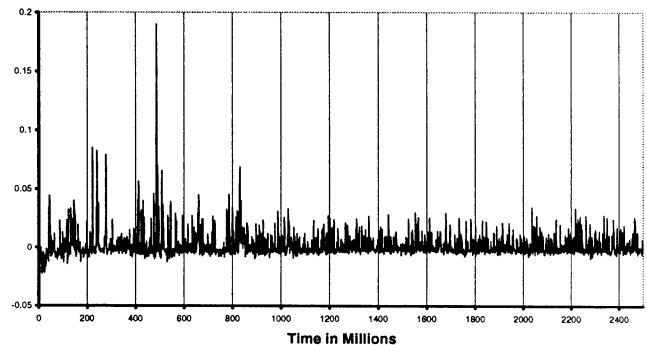
Discussion

The Number index (Figures 2-3) is included only to illustrate the utility of shadow normalization, and is not intended as a contender for a useful measure of evolvability. The Maximum Integral index (Figure 4) is intended to behave similarly to the "adaptive activity" of Bedau and Brown (1999). Rather than plotting the population integral of each genotype, we just plot the envelope over the curves, by plotting only the value of the largest integral at each time step. This visualization still

permits us to see the waves of adaptation. However, it also illustrates a problem with the method: Once evolution has ended, and we enter a period of stasis, we are left with a large integral. Our ideal index would drop to a zero value when evolution has ceased.

The Maximum Proportion index (Figure 5) also reflects the waves of adaptation, in that adaptations cause a transient increase in the frequency of the genotypes which initially bear them. A drawback with this index is apparent in the period between the appearance of the optimal replicator 27aal at t = 800, and the appearance of the 15 byte parasites at t = 1200. During this interval, the space was completely dominated by 27aal causing a high value of the index, although relatively little evolution occurred. The ideal index would have dropped to zero during this period of relative stasis.

One solution to these problems is to use an index which is sensitive to the *rise* in frequency of genotypes bearing adaptations. One such index is the Derivative of Maximum Proportion (Figure 6). This index shows sharp positive spikes at the moments of appearance of new adaptations, yet remains near zero the rest of the time. This behavior approaches that of the idea measure we are searching for. A drawback of this method is that it appears to have a low signal to noise ratio. This problem might be alleviated by integrating under the curve, as the spikes may produce a large integral.

Conclusions

It appears to be possible to quantitatively measure evolutionary activity, in the form of the appearance of new adaptations in a freely evolving system. Some of these measures approach the ideal of having a positive value when adaptation occurs, and a near zero value otherwise. Yet each of the measures tested has its unique drawbacks. Further work is needed to develop the ideal measure of evolvability.

References

M. A. Bedau and C. Titus Brown, "Visualizing Evolutionary Activity of Genotypes." *Artificial Life* 5 (1999): 17-35.

Ray, T. S. 1991. An approach to the synthesis of life. In: Langton, C., C. Taylor, J. D. Farmer, & S. Rasmussen [eds], Artificial Life II, Santa Fe Institute Studies in the Sciences of Complexity, vol. XI, 371-408. Redwood City, CA: Addison-Wesley.

Ray, T. S. 1994. Evolution, complexity, entropy, and artificial reality. Physica D 75: 239-263.

The Control Structure of Artificial Creatures

Zhou Dengyong, Dai Ruwei

AI Lab, Institute of Automation, Chinese Academy of Science

P. O. Box 2728, Beijing (100080), China

Email: *dai@ht.rol.cn.net*

Abstract

The motivation of this article is to integrate some ideas from the science of complexity, behavior-based AI and the theory of Metasynthesis for Intelligence System and to design a computational model to implement these ideas briefly. Our simulated micro-world is a two-dimensional grid inhabited by some resources including food and water, walls, shades, bug and a special creature. This artificial creature will fulfill a set of goals in the complex, dynamic and unfriendly environment. The creature is consisted of a set of self-interested agents, and has the life-like character by means of the interaction between its lifeless agents, besides the interaction between the creature and its environment. The experimental result demonstrates the usefulness of this model, and this is only the first step for our ultimate goal.

1. Introduction

This research is influenced by the science of complexity, bottom-up AI and the theory of Metasynthesis of Intelligent Systems. The motivation of this article is briefly introduced as follows.

At the middle of 1980', in SFI, some scientists realized that the traditional approaches couldn't be used to resolve the complex problems, and summarized a new field named complex adaptive system (Holland95, Holland97). Their research has impacted on the classical scientific research violently, such as economics. The Emergence and evolution are the core ideas of the science of complexity, and obviously Artificial Life (AL) is a remarkable branch of the science of complexity, in which the adaptation is viewed as the important character of life and emphasizes synthesis rather than reduction (Langton 1989, Ray1994). AL has been influencing the development of AI and cognitive science intensively (Maes1993, Steels1995, Solom1997). On the other hand, the conception of life has become abstract since Langton founded artificial life that makes us understand complex adaptive systems more easily and more vividly than before. For example, we can view some economic system, social system or even some part of human body, such as human brain as an artificial life system.

Almost at the same time, behavior-based AI has emerged in the study of artificial intelligence, which is also termed "bottom-up AI" (Brooks1986). A creature or an autonomous agent is viewed as consisting of a set of behaviors. The intelligence of the creature means that it can select the right one from a repertoire of actions at any time so as to achieve its multiple time-varying goals in a dynamical environment. There are so many computational models of action selection which have been proposed by computer scientists or ethologists (Maes1990, Tyrell 1993, Blumberg1994). The ancestors of behavior-based AI and the Society of Mind theory (Minsky1988) are ethology and cybernetics in practice, which are both analogic to Braitenberg's vehicles (Braitenberg1984). The basic idea of behavior-based AI is that the functionality or structure can emerges in the interactions between an agent and its environment, between the different components in an agent and between the agents of a social system (Brooks1991, Raynolds87, Mataric1995, Dorigo1996.). By the way, it is very interesting to evaluate the Society of Mind from the standpoint of complex adaptive systems. We can find that k-line is key to Minskey's theory because it indicates how Minskey's mind develops from simpleness to complexity, or in other words, how to build new blocks (new agents) from those old blocks (old existed agents).

At the end of 1990', Chinese Scientists brought out a new scientific subject founded on a mass of practice in so many fields, which is the theory of Open Complex Giant System (OCGS)(Qian1990). The basic opinion of this theory is that some very complex systems in the nature or our society can be described by OCGS, such as human brain systems, human society systems and economic systems. Then they founded the theory of Metasynthesis of Intelligent Systems (Dai1996), which is a new theory about intelligent system, including artificial systems and nature systems, based on the introspection on classical AI and the theory of OCGS and have been influencing the development of AI and Automation in China intensively. They stress that we should understand intelligence by synthesizing all kinds of theories rather than by a monolithic one and we couldn't separate intelligence systems from their external environment and break intelligence up into some pieces, such as perception, deduction, learning and emotion, and that imaging thinking is the essence of intelligence, rather than logical
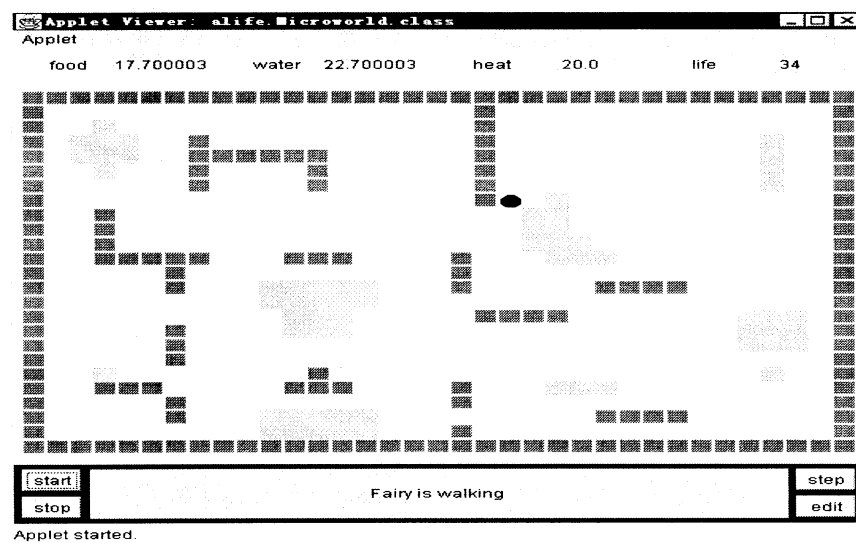
thinking.

It is so surprised for us to find that these ideas are very stupendously consistent in so many aspects. We try to implement these ideas through a simulation world or would-be world (Casti1996). This work is concerned with the control structure of an artificial creature which makes it fulfill a set of goals in a complex, dynamic and unfriendly environment. The creature can sense the environment through its sensors and act on the environment using its effectors. In this article, a creature is viewed as consisting of a set of self-interested agents, including the sensor agents, effector agents, behavior agents and motivation agents and so on. All these agents form a tangled web. The life-like character emerges from the interaction among these lifeless agents and between the creature and its environment. Then the core problem about designing the control structure of artificial creatures is how to construct this web consisting of all kinds of agents. Adding more new agents to this web, the web is becoming more complex and the creature will be more adaptive to its environment.

The paper is organized as follows. Next section introduces an experiment system chosen to implement our ideas. Then the computation model of our artificial creature is presented. Finally, we draw some conclusions and point out the future work.

2. An Experimental System

Behavior-based AI often emphasizes that we should build real robot roaming in the physical environment to explore human intelligence or life. In my opinion, it is not

necessary. A autonomous agent can be embodied in a software environment("softbot")or animation environment (virtual character)too, from which we can learn so much as the same as physical robots. Our micro-world is a two-dimensional grid inhabited by some resources including food and water, walls, shades, bug and our creature. Resources are distributed in the environment, of which the locations initialized when we start the stimulator. At the first time, they will have grown up again after 10 time steps since they are consumed. Then, they will do after 20 time steps. Shades roam in the world randomly. Our creature's temperature will fall if some shade near it. All bugs roam in the environment randomly without hitting walls, too. Although they don't eat every resource, they will bite our creature once they find it near them. A bug's bite will decrease the resources in the creature.

Now let us introduce the protagonist. The goal of the creature are to keep its internal variables in some settled ranges including its food level, water level and body temperature, or else it will be dead. The ranges of its food and water level all are from 0 to 20 units, the resource in one cell meaning one unit. The creature's body temperature can't be too high or too low and should be fallen into this range from 0 to 50 units.

The creature is equipped with all kinds of sensors and effectors. The maximum distance at which the creature can perceive the contents of a square is 5 squares away. All actions that need to be incorporated into the creature's repertoire are the following: (1) Eating. The creature can eat the food near it and increase its food level and body temperature; (2) Drinking. The creature can drink the water near it and increase its water level and decrease its body temperature; (3) Walking. The creature can walk to

any cell only if it is not occupied by something, at the same time its food and water level is decreased and its body temperature is increased; (4) Sleeping. The creature's body temperature will fall if it keep static, but its food and water level will keep changeless; (5) Looking. The creature can perceive the contents of a square at 8 squares away when it is looking.

3. Computational model

At present, the creature is consisting of all kinds of agents, including sensor agents, effector agents, behavior agents and motivation agents. In future, we will add more agents into the agents' web, such as emotion agents.

3.1. Sensor Agents and Effector Agents

Sensor agents sense the outside stimuli and inside variables, such as the sense-food-agent, sense-water-agent, sense-bug-agent, sense-shade-agent, sense-vacancy-agent, sense-food-level-agent, sense-water-level-agent, sense-temperature-agent. Motivation agents and behavior agents make use of the activities of sensor agents. Effector agents perform motor actions on the external world, such as the walk-agent, sleep-agent, look-agent, eat-food-agent, drink-water-agent. Behavior agents exploit effector agents to influence the world.
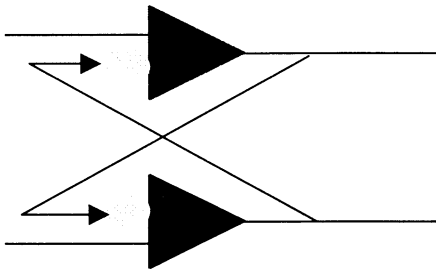
3.2. Motivation Agents

Motivation agents designate the internal states, including motivation-food-agent, motivation-water-agent, motivation-hot-agent and motivation-code-agent. At every time step, every motivation agent computes its strength value based on the internal states and external stimuli. Is the motivation activated which has the maximal strength value? No. Our creature will dither between two motivations and only have opportunistic behavior if we do so. We should make the creature's action have the right amount of persistence. Some computational models for solving this problem have been provided by ethologists

and computer scientist (Minskey1986, Tyrell1993, Blumberg1994). The essence of this kind of model is that the activated motivation will inhabit the others. In this paper, we use this analogous model to implement the persistence of actions while allowing for opportunistic behavior. In future work, we will try to do it by emotion computing. Cross-connected motivation agents construct the creature's motivation system.

3.3. Behavior Agents

Every behavior agent exploits some sensor agents to get environment data and some effector agent to act upon the environment. For example, approach-food-agent will try to approach some cell and activate effector agent walk-agent once sense-far-food-agents is active, and eat-food-agent will try to eat food and activate eat-food- agent when sense-near-food-agent is active.

Self-interested behavior agents are interconnected to some become behavior systems or behavior swarms. Behavior agents in a swarm inhibit or activate each other. Every behavior swarm belongs to some motivation agent. When some motivation is activated, its behavior swarm will excite a favorite behavior. Some behavior agents belong to all swarms, such as walk-agent and avoid-enemy-agent, so motivation system and behavior systems construct a loose hierarchical architecture.
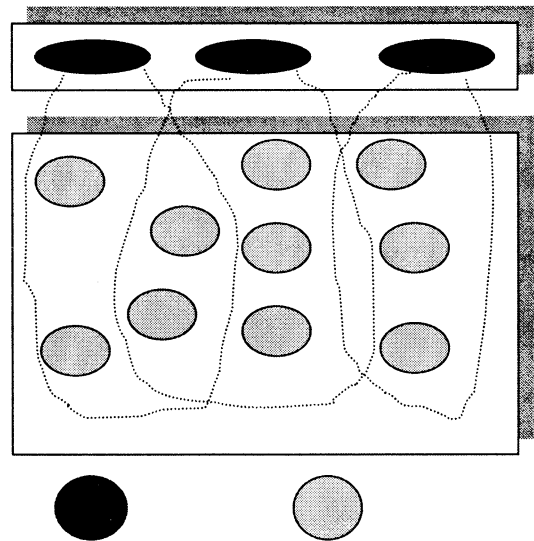
In any behavior swarm, behavior agents are ranged according their priorities that are decided by designers in advance. So the first priority behavior is evaluated, and if it is failed to being executed, the second priority behavior is considered, and so on. At the same time, the higher priority behavior agents can change the priorities of the lower priority ones if they belong to the same swarm. Then the priorities are dynamic among every behavior

swarms.

For example, the motivation-food-agent has its behavior swarm, which is consisted with the avoid-enemy-agent, eat-food-agent, approach-food-agent, look-food-agent, wander-agent, sleep-agent. When the motivation-food-agent is active at some time step, there is a behavior to be selected to control the creature's motor among the corresponding behavior swarm. If no enemies are near the creature, the avoid-enemy-agent can't be executed, and the eat-food-agent is evaluated. If no foods are near the creature, then the approach-food-agent is considered, and so on. When the avoid-enemy-agent is excited, the following agents' priorities can be changed by it. Analogously, the active approach-food-agent can make the wander-agent's priority higher than the look-food-agent's.

4. Conclusion and Future Work

We firstly describe our motivations of this article from three aspects, including the theory of complex adaptive system, behavior-based AI and the theory of Meta-synthesis of Intelligence System. Then we try to implement ideas through a simulating world. A simple control architecture of adaptive artificial creature is proposed, which combines motivation systems with behavior systems to come into being a loose hierarchy. It is proved to be useful in our experiment. All these work is very rough only the first step to our entire goal.

In the near future, we will incorporate emotion agents into the existed agents' web to moderate the relations among motivation system, behavior systems, sensor systems and effector systems so as to create a more tangled but more flexible network for the control of artificial creature. There have been some good works in this field (Cañamero1997, Velásquez1998). At the same time, we will apply this model to our mobile agent system which supports software agents to roaming on the Internet to achieve adaptive distributed computation. Mobile software could make decisions adaptively based on users interests and network resources, such as bandwidth and host load.

5. References

Blumberg, B.1994. Action-Selection in Hamsterdam: Lessons from Ethology. In *Proceedings of SAB94*, 108-117. Brighton, England: MIT Press.

Braitenberg, V. 1984. Vehicles: Experiments in Synthetic Psychology. MIT Press/Bradford Books.

Brooks, R.A. 1986. A Robust Layered Control System for a Mobile Robots, *IEEE Journal of Robotics and Automation*, RA-2, April.

Brooks, R.A. 1991. Intelligence Without Reason. Computers and Thought lecture. In *Proceedings of ICJAI-91, Sidney, Australia*.

Cañamero, D. 1997. Modeling Motivations and Emotions as a Basis for Intelligence Behavior. In Proceedings of Autonomous Agents'97. ACM Press.

Dorigo M., V. Maniezzo & A. Colorni 1996. The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26(1):29-41.*

Casti, J. 1996. *Would-be Worlds: How Simulation Is Changing the Frontiers of Science.* Chinese version, published by Shanghai Scientific & Technological Education Publishing House.

Dai, R.W., Wang, J., Tian, J. 1996. *Metasysthesis of Intelligent Systems*. Zhejiang Science & Technology Publishing House.

Holland, J.H., 1995. *Hidden Order: How Adaptation Builds Complexity,* Helix Books/ Addsion-wesley

Holland, J.H., 1997. *Emergence: From Chaos to Order.* Helix Books/ Addsion-wesley.

Langton, C.G. 1989. Artificial Life. In *Proceedings of an interdisciplinary workshop on the synthesis and simulation of living systems,* edited by Langton, C. . Vol. 6 in the series: Santa Fe studies in the science of complexity. Redwood City, CA: Addsion-Weley.

Maes, P. 1990. Situated Agents Can Have Goals. In Maes, P. ed. *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back.*, 49-70. Cambridge, MA: MIT Press.

Maes, P. 1993. Modeling Adaptive Autonomous Agent. *Journal of Artificial Life*, Vol. 1, Number 1/2.

Mataric, M. 1995. Designing and Understanding Adaptive Group Behavior. *Adaptive Behavior*, Vol. 4, 1.

Minsky, M. 1988. *The Society of Mind*. Simon & Schuster, New York.

Qian, X.S., Yu, J.Y., Dai, Y. W. 1990. *Nature Magazine,* 13(1):3-10.

Ray, T.S. 1994. An Evolution Approach to Synthetic Biology: Zen and the Art of Creating Life. *Artificial Life*, Vol.1, No. 1/2, MIT.

Reynolds, C.W. 1987. Flocks, Herds, and Schools: A Distributed Behavioral Model. *In Proceedings of SIGGRAPH'87, Computer Graphics 21(4).*

Sloman, A. 1997. AI and ALife: response to a journalist's request for overview http://www.cs.bham.ac.uk/~axs/misc

Steel, L., Brooks, R. 1995. *The Artificial Life Route to Artificial Intelligence: Building Embodied, Situated Agents,* Lawrence Erlbaun Associates, Hillsdale, New Jersey.

Tyrell, T. 1993. Computational Mechanisms for Action-Selection. Ph.D.thesis. Center for Cognitive Science, University of Edinburg.

Velásquez, J. 1998. A Computational Frame-work for Emotion-Based Control. In: *Proceedings of SAB'98 workshop on grounding emotions in Adaptive Systems.*

Visual communication in swarms of intelligent robot agents

Jeffrey Johnson
Department of Design and Innovation, The Open University
Milton Keynes, MK17 8QH, England.
j.h.johnson@open.ac.uk

Abstract

The behaviour of intelligent swarm robot
agents is investigated using football. New
mathematics is developed for spatial
relationships to represent the dynamics of the
system. These structures can be used for
tactical and strategic planning. Agents can
perceive and recognise the mathematical
structures. An individual agent can signal other
agents by its relative position in these dynamic
structures. Agents can communicate using
these dynamic structural signals. In particular,
agents may deliberately move in ways which
signal their desire to self-organise with other
agents to achieve particular structural
outcomes and goals.

1. Introduction

The major ideas developed in this paper are:

(1) the behaviour of swarms of intelligent
robots agents can be investigated using
the examples of human and robot football
(2) the spatial relationships between football-
playing agents form recognisable
structures
(3) these dynamic relational structures have a
mathematical representation
(4) these structures and structural events
condition what can and cannot happen
next, and can be used in tactical and
strategic planning
(5) individual agents can perceive these
dynamic structures
(6) an individual agent can signal other agents
by its relative position in these dynamic
structures
(7) agents can communicate using these
dynamic structural signals.

Vision is the predominant means of
communication in swarms of human
footballers. Human players are able to 'read'
each other and know what they will do. They
do this by interpreting particular movements,
and a common 'culture', which means they are
able to 'see things the same way'.

This commonality may be the result of the
players training together, learning the nuances

of each other's body movements, and what
these signals mean for future activity.

These observations can be transferred usefully
to swarms of collaborating intelligent robots.

2. The spatial relationships between agents form recognisable structures

The proposition that spatial relations between
agents form recognisable structures will be
argued by example. Figure 1 shows a common
relational structure, which most readers will
recognise. The defending player must attempt
to halt the progress of the player with the ball
by approaching him. When player 3 gets
sufficiently close, player 1 passes the ball to
player 2, so player 3 is effectively neutralised.
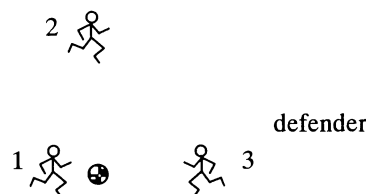
Figure 1. A common relational structure
between football-playing agents
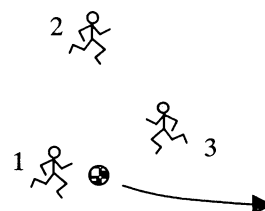
Figure 2. Player 3 is drawn out of position
by the relationship between players 1 and 2.

Of course, player 3 will recognise the
possibility of being 'wrong-footed', and may
attempt to position himself in a such a way that,
if player 1 passes to player 2, then player 3
may either intercept the pass or subsequently
tackle player 2 (Figure 2). The problem with
this is that player 1 may now keep the ball and
dash past the defender. This case is very
interesting because player 2 has drawn player 3
out of position. Although player 2 has not
touched the ball, he has induced player 3 to put
himself in an inferior position.

3. The mathematics of dynamic collaboration relationships

3.1 A fundamental symbolism

Suppose a system has n interacting agents, a_1, $a_2, a_3, \ldots, a_n$. Let R be any relationship on these agents. The relation R could be very complex indeed. For example, the agents might be the grains of sand on a beach, and R the spatial relationships between them. More dynamically, the agents may be the soldiers and their equipment in a battle, and R_t might be the instantaneous spatial relationship between them all at time t.

Let the notation $\langle\, a_1, a_2, a_3, \ldots, a_n \,;R\,\rangle$ represent the set of agents $a_1, a_2, a_3, \ldots, a_n$ under the relation R. The set $\{a_1, a_2, a_3, \ldots, a_n\}$ will be called the *domain* of the relation R.

For example, let $\{\,p_1, p_2, p_3\,\}$ be the set of players shown in Figure 1. Then the configuration shown can be represented by the notation $\langle p_1, p_2, p_3; R \,\rangle$.

Sets such as $\{\,a_1, a_2, a_3, \ldots, a_n\}$ are relatively simple entities, and relatively easy to manipulate. Relations may be very complex entities and very difficult to manipulate. For example, consider all the component parts of a motor car. There are probably some thousands of these in the set of component. Assuming each has a name, these can be listed on a few sheets of paper. The relationship, R, that must hold between these components to give them the emergent property of being a motor car is much more complex. R will probably be composed of very many sub-relations, which in written form will fill the many volumes which make up the assembly manual.

In complex systems it is essential to understand the interplay between sets and relations. Although relations may be very complex, they are *constrained* by their set-theoretic domain. To determine if a relation holds it is necessary to invoke a *pattern recogniser* (Johnson, 1999). Generally building pattern recognisers involves a large investment, and executing a pattern recogniser may involve considerable computation. When dealing with large numbers of interacting patterns, combinatorial explosion may make the computations intractable. Understanding the domains of the patterns and their structure may reduce this computation to tractable levels.

3.2 Representing relations by polyhedra

For simplicity of development, in the first instance, it will be assumed that all the agents perceive each other in the same positions in the same co-ordinate frame. This would be the case for robots receiving position information from some external sensor such as a GPS (Global Positioning Satellite) system used in navigation, or a camera suspended above the pitch in robot football.

Suppose the players in Figure 1 are robots. The three points $P_A = (\,x_A, y_A\,), P_B = (\,x_B, y_B\,)$, and $P_C = (\,x_C, y_C\,)$ can (partially) represent the *state* of the system at time t. The fact that these points are perceived simultaneously imposes a *relation* on them. Let this relation be denoted R_t .

As illustrated in Figure 3, the relationship R_t between the robots creates a structured object with three vertices, which we call a *simplex*. In this case the simplex is a *triangle*. A simplex such as this will be represented by the notation $\langle P_A, P_B, P_C; R_t\,\rangle$, which lists the vertices and establishes the relationship between them.

A simplex with two vertices is a line, which corresponds to the special case of an edge (link) in a graph or a network. A simplex with three vertices is a triangle. A simplex with four vertices is a tetrahedron. In general a simplex with $n+1$ vertices is called an *n-dimensional simplex*, or n-simplex. An n-simplex can be represented by an n-dimensional polyhedron embedded in a multidimensional space.

(a) The 2-simplex $\langle\, P_A, P_B, P_C; R_t\rangle$ is a triangle

(b) The 3-simplex $\langle\, P_A, P_B, P_C, P_D; R_t\,\rangle$ is a tetrahedron
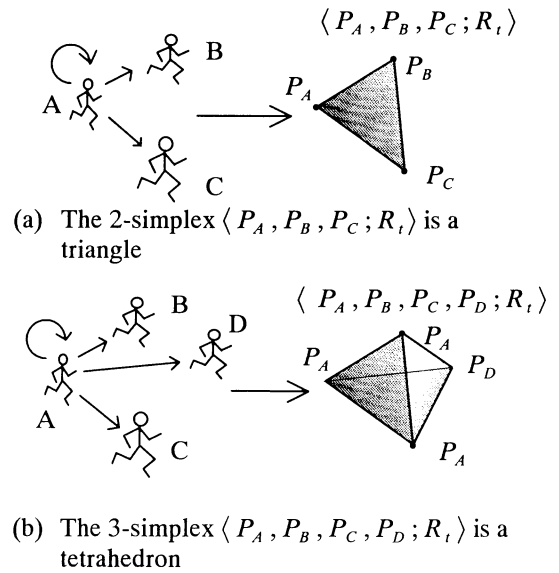
Figure 3.   Simplices determined by the observed relationship between the players

$$R_t \xrightarrow{\hspace{2cm}} R_{t+1} \xrightarrow{\hspace{2cm}} R_{t+2} \xrightarrow{\hspace{2cm}} R_{t+n}$$
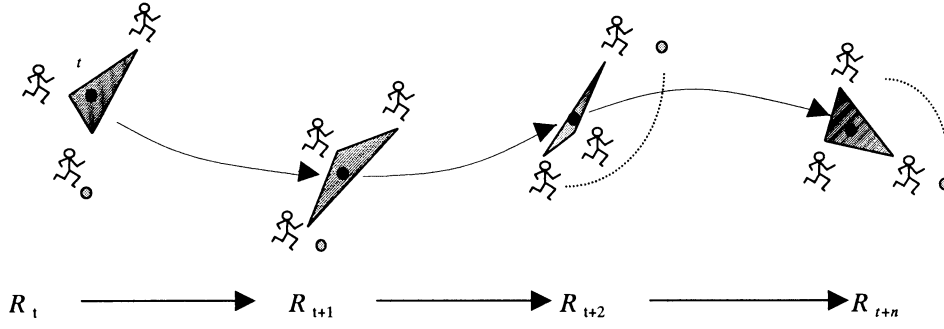
Figure 4. A *polyhedral trajectory* as a trajectory of relational structures through time

Let R_t be the relation between player A and two other players at time t. Then as Figure 4 suggests, the dynamics of football can be represented a sequence of polyhedra, with the polyhedron at time $t+k$ evolving out of that for time $t+k-1$. Let σ_t represent the relational polyhedron at time t. Then the sequence of simplices $(\sigma_t, \sigma_{t+1}, \sigma_{t+2}, \dots, \sigma_{t+k})$ will be called a *polyhedral trajectory*.

The dynamics of multi-agent interactions can be represented by such trajectories. Let $c_x(p_i)$ be the *x-co-ordinate* of player p_i, and $c_y(p_i)$ be the *y-co-ordinate* of player p_i. Then the polyhedron σ_t, possesses the *emergent property* of 'centroid', defined as the point

$$c(\sigma_t) = (c_x(\sigma_t), c_y(\sigma_t)) =$$

$$\left(1/n \sum_{i=0}^{n} c_x(p_i), \ 1/n \sum_{i=0}^{n} c_y(p_i) \right)$$

This concept of centroid allows us to capture the idea of an object moving through space, as illustrated in Figure 5. If the players move *continuously* though time, then the centroid moves continuously through time, forming a smooth trajectory.

3.3 Hierarchical structure

A coherent approach to hierarchical structure is essential to understanding systems of even modest complexity. The generality of hierarchical structure is that one takes a *set* of elements, applies a structuring relation to obtain a new object, and then one gives that new object a *name*. The named object then exists at a higher hierarchical than its component elements.

This is illustrated in Figure 6. The expression 'Configuration 17' represents the structured set of elements. As such it can be the subject of symbolic reasoning, possibly with no reference to the elements from which it is formed.

On the left of Figure 6 there is a *hierarchical cone*, which is defined to be a pair of the form (set, name). In this case the cone is

$$(\{ P_A, P_B, P_C, P_D \}, \text{Configuration 17}).$$

The set $\{ P_A, P_B, P_C, P_D \}$ is defined to be the *base* of the cone

Figure 5. The dynamics of Figure 4 represented as a sequence of structural events, S1→S2→S3→S4.

Configuration 17   *Level N+1*

The hierarchical
naming mapping,
h

$$R_i : \{ P_A, P_B, P_C, P_D \} \rightarrow \langle P_A, P_B, P_C, P_D ; R_t \rangle$$
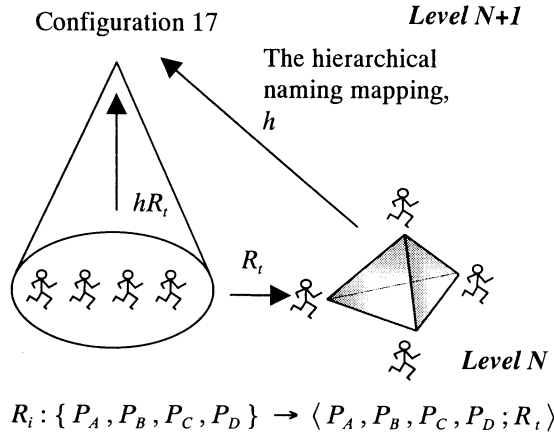
Figure 6. The fundamental diagram of
hierarchical structure

Using this notation, the trajectory in Figure 5
can be viewed as two trajectories, as shown in
Figure 6. If the original sequence of simplices
is said to exist at *Level N*, then the trajectory
made up of a sequence of structures exists at
Level N+1. Conceptually one may consider
the dynamics of football to be sequences of
structures at *Level N+1*, whose details can be
'unpacked' at *Level N* if this is necessary.
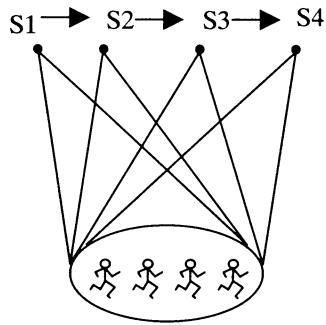
Figure 7. The dynamics of Figure 4
represented by cones with a common base.

The hierarchical representation enables the
distinction to be made between the *continuous*
dynamics of changing relations, and the
discrete dynamics of changing sets.

Figure 7 shows a sequence of structures in
which the underlying set of players is constant,
while the relations on that set evolve through
time producing distinct structures at *Level N*
mapping to distinct vertices at *Level N+1*.

Figure 8 shows a sequence of cones in which
the bases of the cones intersect. This is the
predominant case, in which the structure
evolves with players dropping out, and new
players joining in the relational structure.
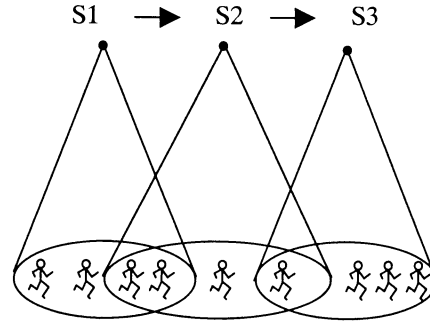
Figure 8. A sequence of cones with
intersecting bases.

Figure 9 shows a sequence in which the bases
of the cones are completely disjoint. This case
might be observed when, for example, a
defending structure S1 passes the ball forward
to an attacking structure S2, when S1 and S2
have no players in common.
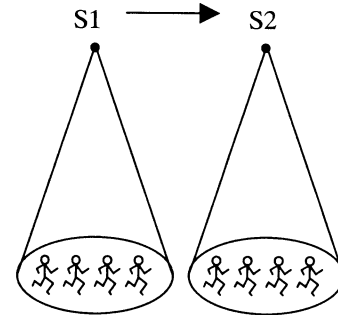
Figure 9. Cones with disjoint bases

3.4 A theory of connectivity

Figure 10 illustrates a fundamental property in
complex systems, namely that of connectivity.
Let the players be labelled left to right *as A, B,
C, D, E, F, G, H, I*. Then define S1, S2, and
S3 by base(S1) = {*A, B, C, D*}, base(S2) = {*C,
D, E, F*}, and base(S3) = {*F, G, H, I*}.

Let these sets, and by extension the structures,
be said to be *q-near* if they share *q+1* elements.
Then S1 is 1-near S2 since they share players
{*C, D*}, and S2 is 0-near S3 since they share
player {*F*}.
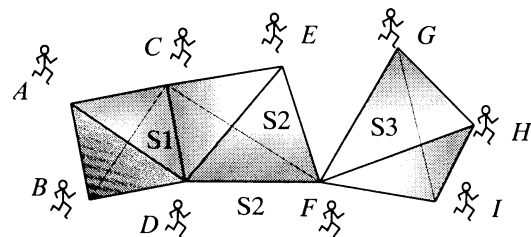
Figure 10. Connected polyhedral structures

I — 23

As can be seen in Figure 10, two simplices are 1-near when they share a line (2 vertices, 1-dimensional), and they are 0-near when they share a vertex (1 vertex, 0-dimensional).

The notation for the simplex $\langle A, B, C, \ldots; R_t \rangle$ will be abbreviated to $\langle A, B, C, \ldots \rangle$ when the defining relation, R, is clear from the context.

When a simplex is represented by polyhedron, any subset of its vertices determines a *face*. Faces can also be considered to be simplices. A set of simplices with all their faces is called a *simplicial complex*.

The q-nearness relation on a set of simplices is a reflexive symmetric relation. Its transitive closure is an equivalence relation which partitions those simplices with dimension q or greater into equivalence classes called *q-connected components*.

In other words two simplices are q-connected if there is an intermediate sequence of pairwise q-near simplices. For example, in Figure 10 the simplex $\langle A, B, C, D \rangle$ is 0-connected to the simplex $\langle F, G, H, I \rangle$ through the intermediate simplex $\langle C, D, E, F \rangle$.

Connectivity through time is clearly important in the evolution of systems. For example, the simplices of the configuration in Figure 9 are disconnected. The implication is that the S1 structure 'passes on' to the S2 structure, and plays no part in the subsequent dynamics (in the short term). To this extent S1 is expendable, and could play a role similar to the sacrifice in chess.

The generality of football dynamics is that each player may be related to many subsets of others players according to different relations.

A set theoretic description of this is illustrated in Figure 11, in which player A belongs to the subsets $\{A, B\}$, $\{A, D, C\}$, $\{A, E, F, G\}$, $\{A, H\}$, $\{A, I\}$, and $\{A, J, K\}$. As the figure suggests, this class of subsets is called a *star*, as is the associated set of simplices with their defining relations (Johnson, 1983).

Thus at any instant of time, each player defines a set of simplices, and therefore a complex. Putting together these structures for all the players forms another simplicial complex which will be denoted K_t. There are $2^{10} - 1$ non-empty subsets of ten players, and therefore 1023 sets of players that footballer A can combine with in Figure 11.
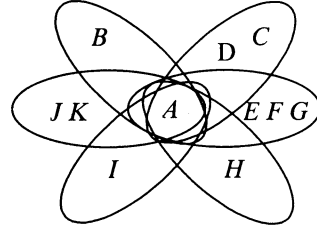
Figure 11. A star configuration of sets containing A.

3.5 Geometrical and topological transitions

With the representation developed in the preceding sections, we can give a general description of a robot football game as a sequence of simplicial complexes evolving through time:

$$K_t \xrightarrow{\varphi} K_{t+1} \xrightarrow{\varphi} K_{t+2} \xrightarrow{\varphi} K_{t+3} \xrightarrow{\varphi} \ldots \xrightarrow{\varphi} K_{t+k} \xrightarrow{\varphi}$$

The transformation φ will be said to be *topological*, or *continuous* if

(T1) for every simplex $\langle A, B, C, \ldots; R_t \rangle$ at time t there is a simplex $\langle A, B, C, \ldots; R_{t+1} \rangle$ at time $t+1$.

(T2) the transformation of the centroid of the simplex is continuous through time.

The transformation φ will be said to be *geometrically discontinuous* if

(G1) there exists a simplex $\langle A, B, C, \ldots; R_t \rangle$ at time t, but no simplex $\langle A, B, C, \ldots; R_{t+1} \rangle$, or

(G2) there exists a simplex $\langle A, B, C, \ldots; R_{t+1} \rangle$ at time $t+1$, but no simplex $\langle A, B, C, \ldots; R_t \rangle$.

If a transformation is geometrically discontinuous it cannot be topologically continuous.

Generally the transformations for the whole complex K_t will always be geometrically discontinuous. However, substructures of K_t may be locally topologically continuous for significant time intervals.

3.6 Structural events

Atkin (1981) suggested that the formation of simplices be considered to be *events* in a system. Thus an event is characterised by a geometric discontinuity between the non-existence of a simplex at time $t-1$, and the existence of that simplex at time t.

In robot football, a set of players may come together to act in consort, and thus initiate an event. As they remain acting together the dynamics of their interaction will be characterised by topological changes, as defined by the centroid of their simplex.

However, any relation R_t will usually be just one of many observable relations between the players.

For example, consider the relation E_t on three players which holds when they form an equilateral triangle. Then Figure 12 shows a transformation which is topologically continuous in terms of the player interaction relation, but geometrically discontinuous in terms of the equilateral triangle relation. Thus the system may be evolving topologically towards a geometric event.
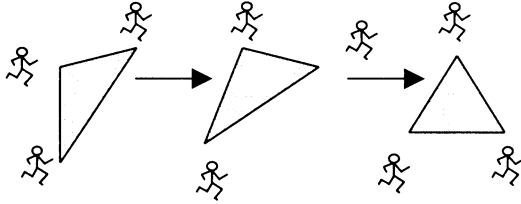
Figure 12. Topological evolution towards a geometric event

A typical geometric event would be one in which there is a suddenly a high likelihood of a goal being scored (Figure 13).

(a)   (b)
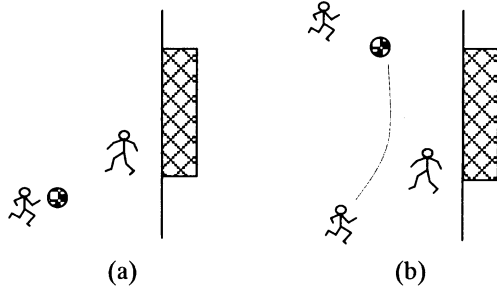
Figure 13. (a) the goalkeeper defends the goal (b) a two-player event likely to lead to a goal

4. Structural events in tactical and strategic planning

One of the subgoals of footballers is to position themselves so that the chance of a goal being scored is high. This means that footballers have to predict possible structures which can evolve from the current structure. A typical subgoal could be achieving a position of the type shown in Figure 13(b).

In this paper the symbol s will be used to mean a goal or a subgoal configuration.

As Figures 14 and 15 suggest, it is possible for very many configurations to evolve from a given configuration, and for many configurations to evolve towards a particular subgoal.
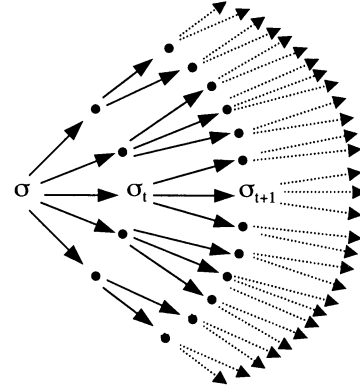
Figure 14. Combinatorially, many events may evolve from a given configuration, σ.
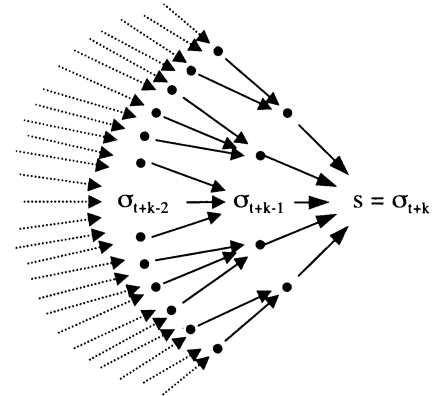
Figure 15. Combinatorially many configurations may evolve to a subgoal, s.

Let $S(\sigma, s) = \langle \sigma_t, \sigma_{t+1}, \sigma_{t+2}, \ldots \sigma_{t+k} \rangle$ be a sequence of configurations which begins with $\sigma = \sigma_t$ and ends with the subgoal, $s = \sigma_{t+k}$. $S(\sigma, s)$ will be called a *trajectory* from σ to s.

Let *tactical plan* be defined to be the decision to adopt a particular trajectory, $S(\sigma, s)$, to move from σ to s.

Given the uncertainties of transitions from one configuration state to another, in highly dynamic systems like football, tactics are only

viable in the short term. Thus, encountering the structure σ may *trigger* the plan to adopt the trajectory $\langle \sigma_t, \sigma_{t+1}, \sigma_{t+2}, \ldots \sigma_{t+k}\ s \rangle$, in order to attain subgoal s starting from σ.

Whereas tactics focus on particular system states, and attempt to *control* the system to stay on a particular trajectory, strategy concerns the dynamics of navigating the space of trajectories.

While pursuing a particular subgoal on a particular trajectory, if the sequence is broken then often the tactic is lost. For example, when a player loses the ball, the current tactic must be abandoned.

We will say a trajectory S *jumps* to a trajectory S' when S is the planned trajectory and S' is the trajectory actually executed, where the S and S' have common configurations at the beginning, but different sub-goals at the end.

For example, Figure 16 illustrates the possibility of trajectory $\langle \sigma, a, b, c, g, d, s' \rangle$ jumping to the trajectory $\langle \sigma, a, b, c, g, h, s \rangle$.

Let s be the subgoal of S and s' be the subgoal of S'. Then if S is predisposed to jump onto S', it is highly desirable that s' is also an acceptable subgoal. Thus the trajectories must be connnected (Figure 17).

Level N+1
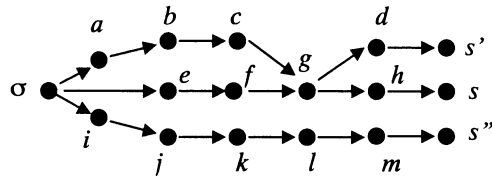
Figure 16.   Three connected trajectories

Strategy involves selecting *collections* of trajectories which have many desirable outcomes, and which jump to other trajectories with desirable outcomes.
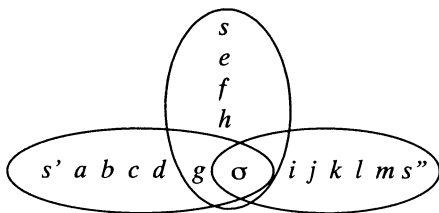
Figure 17.   Connectivity between plans

In particular, it is desirable to chose strategies in which the trajectories jump back and forth between desirable outcomes. This is necessary

for longer-term objectives to be planned and achieved. In a sense one wants the trajectories to be contained with an *envelope*. To achieve this the trajectories must be connected appropriately.
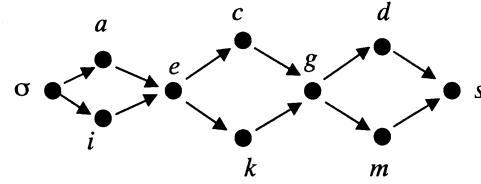
(a)   the trajectory $\langle \sigma, a, e, c, g, d, s \rangle$ is 3-connected to $\langle \sigma, i, e, k, g, m, s \rangle$ through the 3-dimensional face $\langle \sigma, e, g, s \rangle$
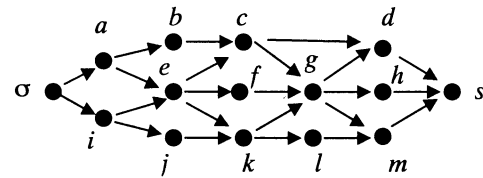
(b) there are 30 paths between σ and s, forming a highly connected complex

Figure 18.   Trajectories between states and subgoals may be highly connected, making it easier to 'stay on track'.

Figure 18(a) shows two trajectories which are 3-connected. In fact there are eight trajectories between σ and s, and the underlying connectivity makes it possible to jump from one to the other.

Figure 18(b) shows a highly connected set of thirty trajectories between σ and s. These trajectories are contained within an envelope which ensures that, even if the particular trajectory jumps, the desired subgoal s will be attained.

Strategic planning can be viewed as the construction of search spaces which behave in well-behaved ways. In searching for optima, there may be no guarantee that any particular state will be sustained, but there should be a guarantee that the play will be restricted to a good part of the space.

In robotics, at any time an autonomous robot has to ask itself "what should I do next". This can be at

- the strategic level of asking "what should be my next (sub)goal", or
- the tactical level of "what should I do to achieve an established (sub)goal".

The problem of action planning for football-playing robots can therefore be seen to be the problem of *generating* possible alternatives, and *selecting* between those alternatives. But, in the light of this discussion it can be seen as more than this. The problem becomes that of generating possible alternatives, and selecting strategic *subsets* of alternatives according to their connectivity properties.

5. Individual agents can perceive these dynamic structures

5.1 Perceiving the structures
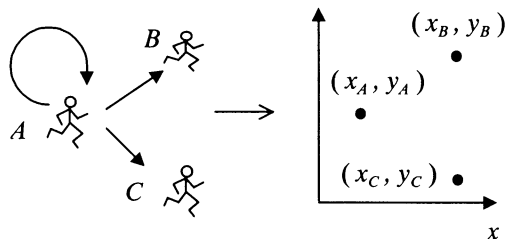
Figure 19. Football player A uses vision to abstract his and other players' positions.

The structures discussed in the previous sections have implicitly assumed that the positions of the robots are known. In practice this means using a vision system to map the robots to co-ordinates (Figure 19).

5.2 Pattern recognition and machine vision

Machine vision is an under-estimated problem in robotics. In the first instance we will suppose that a camera above the pitch can deliver x-y co-ordinates for everything, and that these two-dimensional data are equally available to all the robot agents.

Given the position data of the atomic objects such as the robots and ball, it is necessary to recognise the kind of structural configurations discussed in this paper.

Given a team with eleven players, there are 2^{11} minus one (2047) non-empty subsets:

> 11 with one member
> 55 with two members
> 165 with three members
> 330 with four members
> 462 with five members
> 462 with six members
> 330 with seven members
> 165 with eight members
> 55 with nine members
> 11 with ten members
> 1 with eleven members

Many relations may be imposed on these to produce various configurational patterns.

Two major research questions are therefore:

(i) of the many combinatorial configurations of n robots, which are interesting, how can they be usefully classified for tactical and strategic planning?

(ii) How can these patterns be recognised automatically?

5.3 Skeleton polyhedral trajectories

It is unlikely that agents or robots will be aware of all the combinatorial possibilities available from given starting position. As Figure 20 suggests, an agent may 'feel' that σ_5 can follow from σ_1, without working out precisely how to get there. Thus agents may use *skeleton trajectories* in their planning.
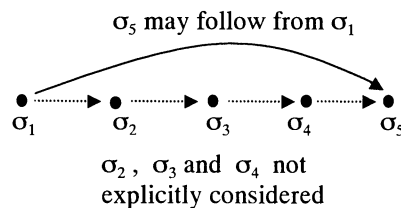
Figure 20 A skeleton polyhedral trajectory

5.4 Cognitive dissonance between robots

When robots do not have a global worldview supplied by an external camera, they must use their own vision systems to form a worldview. The nature of machine vision is such that it is (currently) extremely unlikely that two different robots will perceive the world in exactly the same way, even supposing their co-ordinate frames can be aligned.

Put simply, robots construing the world through their own vision systems will usually see it differently. We will call this *cognitive dissonance* between *the robots*.

To a greater or lesser extent, cognitive dissonance is the norm. This means that if one robot sees some future configuration as desirable, another robot may at best see a similar configuration. This is not surprising. Human football players cannot locate each other with great precision, for them it is sufficient that the other players are 'more-or-less' in the right place for a structure to hold. This again emphasises the need to represent *classes* of configuration patterns as 'similar'.

6. An agent can signal others agents by its relative structural position

6.1 Body-language robot communication

Human beings communicate extensively through the ways they hold their bodies. Although such *body language* may be considered in terms of static attributes such as a raised hand or a smile, body language can also be dynamic. For example, the wave of a hand, or the transition from a frown to a smile may communicate something different to a hand held stationary or a fixed smile.

Although robots such as the 8 cm cubes which play the Mirosot version of robot football may not be able to articulate parts of their bodies for communication, they can certainly communicate through moving position.

This is analogous to humans dancing, in which the movements of the body through space relative to other humans may have particular interpretations.

Thus we can postulate even the simplest robots might communicate through body language.

In this paper we have gone one step further. Certainly it is asserted that robots can communicate through body movements. But we also postulate that those body movements are purposeful in forming desirable structures on trajectories resulting in the attainment of useful subgoals.

6.2 Communication by movement which signal possible structures

Finally we come to the central idea of this paper. The idea is that an agent may 'see' that the present state of the system could evolve towards some desirable state. More formally, the agent 'sees' that subgoal s can follow from the current system state σ by some trajectory $S(\sigma, s)$.

It is unlikely that the agent will see every transition in any particular $S(\sigma, s)$, but suppose agent A 'sees' that some intermediate σ' will clearly signal the $S(\sigma, s)$ postulate.

So, suppose that agent moves in such a way that other agents become aware that σ' is a possible structure, and that agent A is deliberately moving itself to make σ' happen.

One possibility is that another agent, B, may follow a line of reasoning such as

"why is A moving so purposefully like that? That movement could lead to σ', σ'', ... etc. Ah! σ' may lead to the desirable subgoal s. I see! Agent A is suggesting that we collaborate to form the trajectory $S(\sigma, s)$. I agree it's a good idea. I'll move in such a way as to communicate to A that I have understood its message, and that I am prepared to collaborate in the plan".

In response to this agent A might reason "Good! Agent B is acting in a way which is consistent with trajectory $S(\sigma, s)$. I'll reinforce this understanding by moving as follows ..."

Although there is plenty of scope for misunderstandings, one can imagine that once two agents have 'locked on' to a particular trajectory, that there is rapidly escalating positive feedback until both are certain of the intentions of the other.

6.3 Neural communication

In the previous section the communication between the agents was postulated to occur through reasoning from the visual evidence of the other's actions. However, humans often act before they are aware that they are engaging in any logical processing. The visual data comes in and they just act.

This could be called the 'neural processing' response to an agents movement communication. If two or more agents have practised a composite move many times, then any antecedent movement by any of them may trigger recognition of that move.

For example, in football one player may see a colleague take the ball in a certain way, and immediately begin running 'into a space'. By so doing it is clear to the other agent that he wants the ball to be passed. Perhaps he reasons "he's moved into a space, therefore he wants the ball". But perhaps the perception of the space immediately triggers the recognition that the other agent wants the ball.

6.4 Logical and neural processing are complementary

Logical and neural communication are complementary rather than competitive. In the book "What computer (still) can't do" Dreyfus argues that human computation may progress

from explicit symbolic processing to implicit neural processing as expertise increases.

Communicating by movement and vision can use either mode of computation, and may reflect the Drefus' view of intelligence and learning.

7. Agents can communicate using visual signals alone.

Human footballers seem to manage perfectly well with visual communication by itself. Certainly they augment visual communication by shouting, but whether this conveys more information and helps to relieve tension and/or exaggerate their expressions is debatable.

In robot football, extensive use is often made of radio communication. One of the speculations in this paper is that this is not strictly necessary. From a scientific viewpoint, this postulate can be tested by investigating robot football systems which do not use radio communications.

From an engineering perspective, there is no reason not to use radio communications to augment the visual communication.

To test the ideas developed in this paper it is proposed to conduct the following experiments:

[1] Collect data on a number of robot football games, and investigate the polyhedral representation, and polyhedral trajectories.

[2] Abstract a set of configurations which occur relatively frequently, and which have relatively good or bad outcomes.

[3] Create pattern recognition systems to detect those configurations automatically

[4] Build decision-making functionality into robots to make them autonomously decide that they want to follow some particular trajectory

[5] Build movements into the robots which will be recognised by other robots, and which will trigger collaborative behaviour.

[6] Investigate the convergence of collaborating robots towards some trajectory selected by one or more robots, and communicated to the others by their movements.

Preliminary work has been done on these ideas with colleagues at the University of Girona in Spain, and also colleagues at the University of Plymouth in England.

It is planned to continue this research at Oita University in Japan in the spring of 2000, and hopefully we will have results to report later in the year.

8. Conclusions

This paper has investigated the following ideas:

- the behaviour of swarms of intelligent robots agents can be investigated using the examples of human and robot football
- the spatial relationships between football-playing agents form recognisable structures
- these dynamic relational structures have a mathematical representation
- these structures and structural events condition what can and cannot happen next, and can be used in tactical and strategic planning
- individual agents can perceive these dynamic structures
- an individual agent can signal other agents by its relative position in these dynamic structures
- agents can communicate using only these dynamic structural signals.

So far these ideas and suggestions are mainly theoretical. Some practical work has been done, and a series of experiments has been defined which should allow the ideas to be tested in the Spring of 2000, with results to be reported thereafter.

References

Atkin, R.H., *Mathematical structure in human affairs*, Heinemann Educational Books, (London), 1974.
Atkin, R. H., *Multidimensional Man*, Penguin Books (London), 1981.
Casti, J., *Complexfication*, Abacus Books, (London), 1994.
Dreyfus, H. L., *What computers(still) can't do*, MIT Press,(Cambridge, Mass), 1994.
Johnson, J.H., 1983, 'A theory of stars in complex systems', *Planning and Design*
Johnson, J.H., 1999, 'The design and control of self-organising agent systems', 2nd Congres Catala d'Intel-ligencia Artifcial, 25-27 Oct 1999, J. L. de la Rosa (ed), ACIA, Spain.

Evolutionary On-Line Self-Organization of Autonomous Robots

Dario Floreano and Joseba Urzelai
Laboratory of Microprocessors and Interfaces
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland

Abstract

We review recent experiments in evolutionary robotics carried out in dynamic environments and across different robotic platforms. We then introduce a new evolutionary approach where robots are evolved for their ability to adapt online. Several experiments show that this new approach is much faster, more powerful, and scalable than the traditional approach.

1 Evolutionary Robotics

Autonomous robots are largely replacing computers as a metaphor for investigating natural and artificial intelligent systems because they interact with a real environment through sensors and actuators in a closed feedback loop, are subject to the laws of physics, operate in real-time, and are required to cope with partially unknown and unpredictable situations. Artificial evolution is a selectionist procedure that discovers suitable controllers by exploiting the interactions between the robot and its environment rather than following a model-based adaptation scheme [1]. The approach is characterized by online evolution carried out on physical robots without human intervention and simple fitness functions in order to emphasize environmental interactions (figure 1).

Figure 2: An evolved 4 legged robot. The control system of the robot, its body size, and length of legs have been evolved in realistic 3D simulations. The physical robot in the picture has been partially built according to the evolved genetic specifications. The evolved control system is transferred from the simulated to the physical robot. Such evolved robot can walk and avoid obstacles. The robot is approximately 20 cm long and weights less than 1 kg without batteries. Leg control is given by low consumption HC11 microcontrollers.
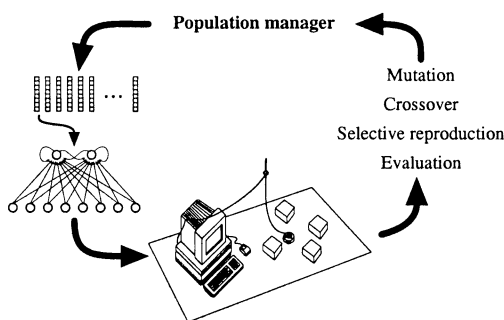
Figure 1: A single physical robot is connected to a host computer through a serial cable with rotating contacts. The serial cable provides power to the robot and data communication. The population of chromosomes, genetic operators, and decoding takes place on the host computer, but the decoded control system runs on the onboard processor. The fitness is computed onboard without external measuring devices.

Evolved sensory motor controllers adapt their navigation strategies to the physical characteristics of the environment and of the robot hardware. The methodology has been applied to several types of robots, with wheels and legs (figure 2). When placed in more complex environments, robots can evolve neural mechanisms that build internal representations of space and time in relationship to internally-defined goals [2].

When co-evolved with a competing robot (figure 3), the reciprocal bootstrapping of the competing controllers drives the ecosystem to increased levels of com-
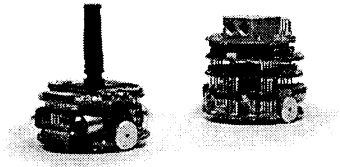
Figure 3: Prey and predator robots co-evolved in competition with each other. The predator on the right has a vision system; its fitness is inversely proportional to the *time* it takes to hit the prey. The prey can be seen thanks to its characteristic black protuberance and does not have vision, but it can detect if something is nearby with infrared sensors and can go at double speed than the predator. The fitness of the prey is proportional to the time it manages to survive without being hit by the predator. The "artificial brains" of the two robots are artificial neural networks.

plexity and eventually to behavioral cycles displaying rapid alternation of non-trivial pursuit-evasion strategies [3].

The most important concept is that the fitness function should leave space for free interaction between the robot and its environment; in other words, the fitness function should not be very detailed. This allows the robot to explore several different ways of solving a problem making evolution easier, faster, and often surprising for an external observer.

2 Evolution of Adaptive Robots

Traditionally, artificial evolution operates on parameters of the controllers, such as synaptic connections and architectures, that are maintained fixed during operation of the controller. This approach does not capture the adaptive plasticity that characterizes biological nervous systems. One way to re-adapt to new conditions, such as a new robot platform (figure 4), is to incrementally continue evolution in the new conditions, but this often takes long time [4].

I suggest to exploit artificial evolution for discovering adaptive controllers that can continuously modify their synaptic parameters in relation to environmental inputs according to evolved adaptive rules. In other words, the genetic string encodes only the parameters of Hebbian plasticity that drive synaptic modification, but *not* the synaptic strengths. Every time an individual is born, its synaptic values are *randomly initialized*
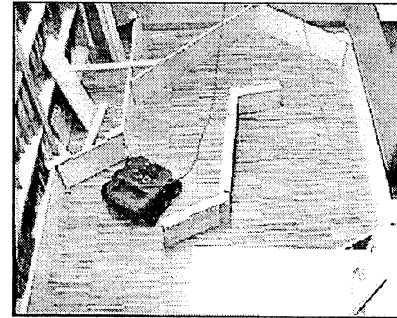
Figure 4: Incremental evolution across platforms. Initially a neural control system has been evolved using a miniature mobile robot Khepera. After 100 generations, the last population has been transferred on the much larger and more powerful Koala robot shown in the figure and evolution has been resumed. The population takes 20 generations to readapt to the new sensors, motors, and geometry of the larger robot.

Condition	Bits for one synapse				
	1	2	3	4	5
1	sign	strength			
2	sign	Hebb rule		rate	

Table 1: Genetic encoding of synaptic parameters. 1: Traditional approach; 2: Evolution of adaptive synapses.

(always, from generation 1 to the final generation) and are let free to adapt using the evoved Hebbian rules while the robot moves in the environment.

In this new approach artificial evolution selects individuals that can adapt continuously and online starting always from random initial weights. This does not allow evolution to impress on the synaptic weights a strategy that fits a particular environment (which would not generalize to environmental changes), but rather forces evolution to discover individuals capable of solving a problem by adapting online to the actual environmental characteristics.

Table 1 shows the difference of synapse encoding between the traditional approach (row 1) and our new approach (row 2). In both approaches, the first bit of each synapse encode its sign (excitatory or inhibitory). In the traditional approach, the remaining four bits encode the synaptic strength as a value in the range [0, 1]. No changes take place during the life of the individuals. In the second case instead, two bits encode four Hebbian rules and the remaining two bits the learning rate (0.0, 0.337, 0.667, and 1.0). The four Hebb rules are: "pure Hebb" whereby the synaptic

strength can only increase when both presynaptic and postsynaptic units are active, "presynaptic" whereby the synapse changes only when the presynaptic unit is active (strengthened when the postsynaptic unit is active, and weakened when the postsynaptic unit is inactive), "postsynaptic" whereby the synapse changes only when the postsynaptic unit is active (strengthened when the presynaptic unit is active, and weakened when the presynaptic unit is inactive), and "covariance" whereby the synapse is strengthened if the difference between pre- and post-synaptic activations is smaller than a threshold (half the activation level, that is 0.5) and is weakened if the difference is larger than such threshold. After decoding a genotype into the corresponding controller, each synapse was randomly initialised to a value in the range [0, 1] and modified at each time step according to the corresponding hebbian rule and learning rate.
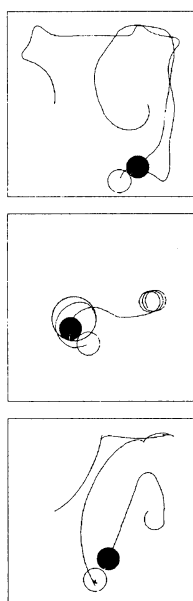
Figure 5: Three examples of co-evolved predator (filled disk) and prey (empty disk). The genes of the two robots encode adaptive characteristics of the synapses. Prey robots always attempt to modify weight in random fashion to generate unpredictable behaviors. Instead predator robots evolve combinations of Hebbian rules that almost always succeed at hitting the prey by quickly modifying their behaviors online.

Evolved adaptive controllers are capable of quickly generating stable behaviors from randomly initialized synaptic strengths. Although the synapses keep changing in relation to presynaptic and postsynap-

tic activations, the controller is dynamically stable. The ability of rapid online adaptation proves useful in dynamic environments. For example, co-evolved predator robots with adaptive controllers are better at catching prey robots because they can rapidly switch between different behavioral strategies depending on the prey behavior [5].
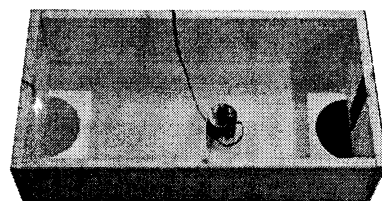
Figure 6: A robot equipped with a vision module can gain fitness points only when it is sitting on the light (grey zone on the left) *when the light is on*. Initially the light is off, but the robot can switch it on by going over the black area on the right. No fitness points are given for the light switching behavior.

Evolution of adaptive controllers can develop solutions for complex problems that the traditional approach can hardly manage. Consider for example the robot in figure 6 equipped with a vision module, proximity sensors, and light sensors. This robot can gain fitness points only when it sits on the grey area on the left *when the light is on*. At the beginning of its life the light is off but it can be switched on if the robot goes to the right area on the right of the arena. Therefore, in order to receive fitness points this robot must evolve the ability to find the light switching area, go there, and once the light goes on rapidly move on the fitness area and remain there for the rest of its life. [1] Our adaptive approach can solve generate very quickly neural controllers that solve this problem in a very reliable and efficient manner, whereas the tradiotnal approach takes almost twice as many generations and the result is a much less efficient (because it is equivalent to a fixed navigation pattern largely independent of the sensory information).

Since in the new adaptive approach synaptic weights adapt online, the genetic encoding can be made more more compact by specifying only the adaptive properties of entire neurons. Very recent results indicate that this amounts to faster evolution and generates more robust controllers [6]. It is thus a promising step toward artificial evolution of developmental rules

---

[1] Note that the robot cannot see the patches on the floor and therefore must look at the patterns on the walls and at the light.

for neural morphologies. From an engineering perspective, evolution of adaptive controllers provides a method for generating systems capable of rapid self-configuration and increased robustness.

Acknowledgements

References

[1] S. Nolfi and D. Floreano. *Evolutionary Robotics: Biology, Intelligence, and Technology of Self-Organizing Machines.* MIT Press, Cambridge, MA, in the press.

[2] D. Floreano and F. Mondada. Evolution of homing navigation in a real mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics-Part B,* 26:396–407, 1996.

[3] D. Floreano and S. Nolfi. God Save the Red Queen! Competition in Co-evolutionary Robotics. In J. Koza, K. Deb, M. Dorigo, D. Fogel, M. Garzon, H. Iba, and R. L. Riolo, editors, *Proceedings of the 2nd International Conference on Genetic Programming,* San Mateo, CA, 1997. Morgan Kaufmann.

[4] D. Floreano and F. Mondada. Evolutionary Neurocontrollers for Autonomous Mobile Robots. *Neural Networks,* 11:1461–1478, 1998.

[5] D. Floreano, S. Nolfi, and F. Mondada. Co-evolution and ontogenetic change in competing robots. *Robotics and Autonomous Systems,* in the press, 1999.

[6] D. Floreano and J. Urzelai. Evolution of Neural Controllers with Adaptive Synapses and Compact Genetic Encoding. In D. Floreano, J-D. Nicoud, and F. Mondada, editors, *Advances in Artificial Life.* Springer Verlag, Berlin, 1999.

Prediction of Significant Oculocardiac Reflex by Nonlinear Complex Cardiac Dynamics

Myung-Kul Yum M.D., *Hee-Soo Kim, M.D

Dept of Pediatrics, Hanyang University Kuri Hospital, 249-1 Kyomundong, Kuri-shi, Kyunggi-do, 471-701, Korea. mkyumm@email.hanyang.ac.kr

*Dept of Anesthesiology, SNUH, #28 Yongon-dong, Chongro-gu, Seoul, 110-744, Korea, dami0605@snu.ac.kr

This study was aimed to determine whether preoperatively measured linear and nonlinear analysis of heart rate dynamics might predict the occurrence of oculocardiac reflex (OCR) during the strabismus surgery in children (n=132, 7.12 years ± 3.01). We compared time- and frequency-domain, and nonlinear dynamic indexes of preoperatively measured RR interval data between the OCR positive (maximum heart rate decrement > 20 beats per minutes during the traction of extraocular muscle, n=54) and negative groups (< 20 beats per minute, n=78). pNN50 (p = 0.0216), rMSSD (p = 0.0021), high-frequency power (p = 0.013), fractal dimension (p = 0.048), and nonlinear prediction error (p = 0.0048) were significantly lower in the OCR positive group than in the OCR negative group. Discriminant analysis using these indexes can identify preoperatively all but two OCR positive patients (52/54, 96.3%). Some preoperatively measured indexes of linear and nonlinear heart rate variability, especially when used in combination, are valuable for predicting significant bradycardia during strabismus surgery in children.
Keywords: Reflex; Heart; Analysis.

Introduction

Severe bradycardia was frequently reported during strabisumus surgery, especially in children [1]. Therefore, to take effective preventive or therapeutic measures to counter the probable forthcoming dangerous bradycardia [2]. However, there has been no effort to find the preoperative clinical or laboratory indicators having predictive value of the bradycardia. It has been documented that the time- and frequency-domain indexes of heart rate dynamics can be of value in predicting cardiac morbidity and mortality [3] and nonlinear indexes measuring complexity [4] or determinism [5] are helpful in the prediction. We are to test our hypothesis that some preoperatively measured linear and nonlinear parameters of heart rate dynamics or their combination may be the predictors of significant OCR during eye surgery.

Method

With the approval of the Institutional Review Board and informed consent, the investigation was carried out on 185 patients (3-15 years). A day before surgery, lead II ECG was monitored and stored in PC. Anesthesia was maintained with enflurane and N_2O. An extraocular muscle was tracted by force at 300gm. During the traction, the minimal heart rate was recorded. If the heart rate decreased more than 20 beat per minute (bpm) than the basal heart rate, traction was released and we injected atropine. The basal heart rate was determined by preoperative ECG data. Maximum heart rate decrement was defined by subtracting the minimal heart rate recorded from the basal heart rate. The patients were classified as OCR positive (maximal heart rate decrement > 20 bpm) and negative (maximal heart rate decrement < 20 bpm). Power spectral analysis was performed by fast Fourier transformation [6]. We calculated the approximate entropy [7] and fractal dimension [8] and nonlinear prediction error [9]. Fig.1 shows plots of calculating three nonlinear indexes from two groups. Differences in the means of linear and nonlinear parameters between the two groups were analyzed with unpaired t-tests. To test how correctly the parameters can

discriminate each group, we used discriminant analysis. All analyses were performed using an alpha level of 0.05 as the criterion for statistical significance.
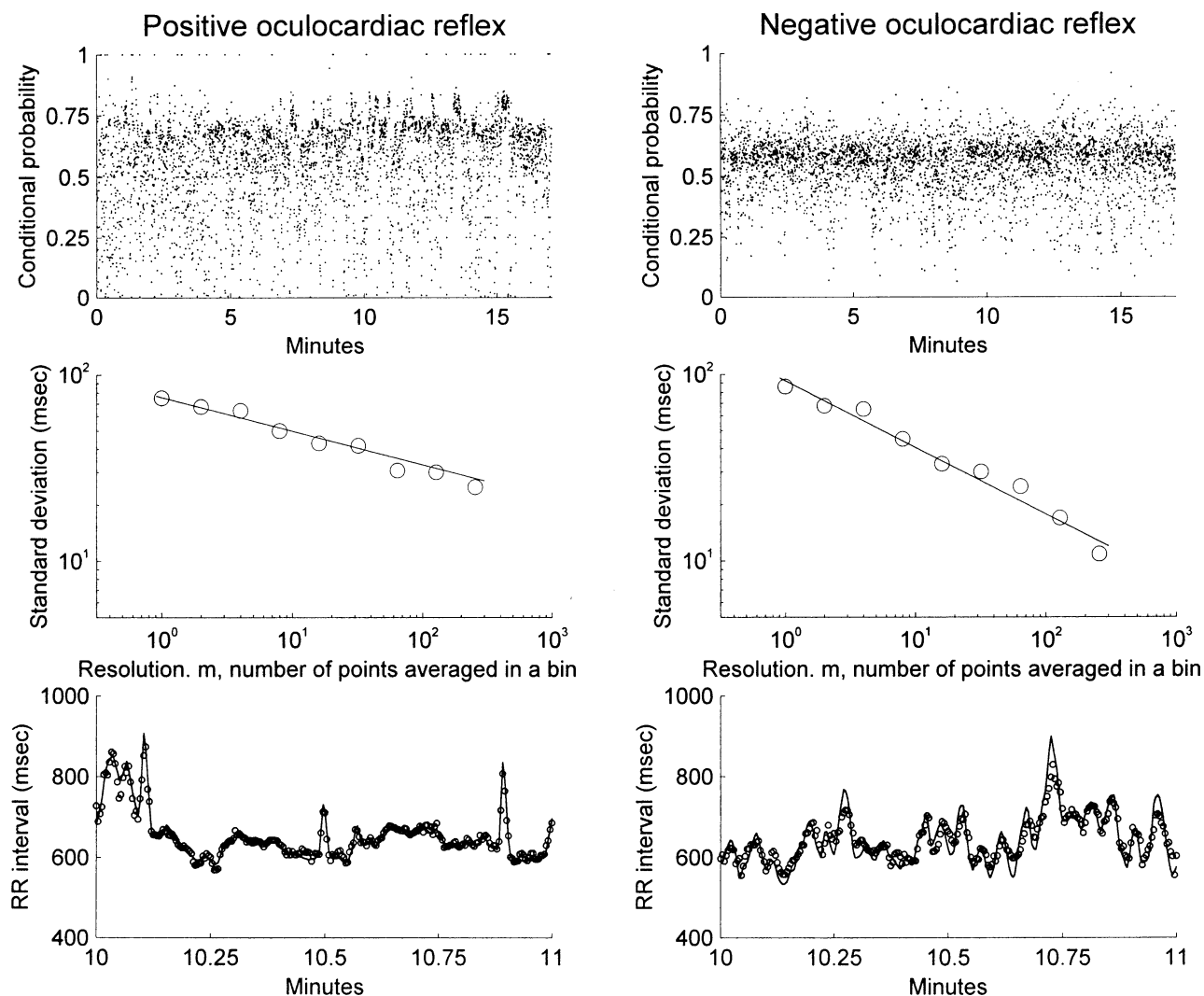
Fig. 1. Plots to show procedure for calculating three nonlinear indexes   Top: conditional probabilities for calculating the approximate entropy, which is the negative of the average logarithmic conditional probabilities. The approximate entropy was not significantly different between the two patients. Middle: The $\log_{10}$-$\log_{10}$ plot of dispersion (standard deviation) against number (m) of neighboring elements for successive averaging. The fractal dimension, which was 1 minus the slope of the plot, is smaller in the OCR positive patient. Bottom: simultaneous plot of the actual RR interval values and predicted ones. The prediction error is the difference between the actual values (line) and predicted values (circle). It is significantly lower in the OCR positive patient.

Results

53 patients were excluded from the study because of arrhythmia. There were 78 (59.1%) in the OCR negative group and 54 (41.9%) in the OCR positive group. Maximal heart rate decrement after traction of the extraocular muscle was 25.7 bpm ± 0.7 (22 - 42 bpm) in the OCR positive group and 12.7 bpm ± 0.5 (-10 - 18

bpm) in the OCR negative group (p = 0.0001). Results of the linear and nonlinear analysis of preoperative measured RR interval data are summarized in Table 1. pNN50 , rMSSD, high-frequency power, fractal dimension, and nonlinear prediction error were lower in the OCR negative group. Table 2 shows the results of discriminant analysis by the model built up by the above parameters which shows statistical significance between the two groups.

Table 1. Preoperatively measured linear and nonlinear heart rate parameters

	OCR negative (n=78)	OCR positive (n=54)	Arrhythmia (n=53)
Mean RR interval(msec)	617.8 ± 124.7	590.5 ± 99.8	604.6 ± 118.6
SD (msec)	51.0 ± 20.7	43.4 ± 12.0	42.8 ± 14.5
rMSSD (msec)	15.2 ± 7.7	10.0 ± 3.0†	11.9 ± 4.9
pNN50 (%)	1.38 ± 2.55	0.17 ± 0.18*	0.54 ± 1.03
LFP (msec^2)	1590.2 ± 1500.5	1044 ± 809.1	1205.5 ± 954.5
HFP (msec^2)	1473.7 ± 1650.0	519.5 ± 1003.2*	789.6 ± 761.1
ApEn	0.80 ± 0.12	0.76 ± 0.14	0.78 ± 0.14
FD	1.18 ± 0.04	1.11 ± 0.03*	1.15 ± 0.07
PE (msec)	14.2 ± 7.2	9.8 ± 3.9†	11.3 ± 4.2

SD: standard deviation, rMSSD = root mean square of all successive RR interval differences, pNN50 = percent of differences between adjacent RR intervals greater than 50 msec, LFP = low-frequency power, HFP = high-frequency power, ApEn = approximate entropy,   FD = fractal dimension, PE = nonlinear prediction error.

Data are represented as mean ± SD. * p < 0.05, †p<0.005 compared with OCR negative group.

Table 2. Numbers of observations and percent classified into OCR positive and negative groups.

From	To	OCR negative	OCR positive	Total
OCR negative		47 (60.2%)	31 (39.8%)	78 (100%)
OCR positive		2 (3.7%)	52 (96.3%)	54 (100%)
Total		49 (37.1%)	83 (62.9%)	132 (100%)

Discussion

Major findings of this investigation were that, preoperatively measured time-domain measures, high-frequency power, nonlinear prediction error and fractal dimension of RR intervals were significantly reduced in the OCR positive group compared to those in the OCR negative children. When the patients were preoperatively evaluated with these measures, there was little chance of failure to identify the OCR positive patients.

Arnold, et al. [10] suggested the association of the profound response to carotid sinus massage with the OCR during eye surgery. However, using the carotid sinus massage maneuver as a predictor of OCR in children because the carotid sinus massage is difficult and not effective in children. The results indicate that preoperative cardiac vagal activity decreased in OCR positive group. The decrease in preoperative nonlinear prediction error and fractal dimension within the OCR positive children is a new finding. Theoretically, the complex and less-deterministic system responds with pliability to external and internal stress and maintains its integrity [11]. Therefore, we expected that the children with higher determinism (lower nonlinear prediction error) and lower complexity (lower fractal dimension) would have an increased possibility to show significant bradycardia when cardiovascular stress with severe vagal stimulation by traction of the extraocular muscle was

present. From the foregoing discussion, we can give some suggestions about the methodology to prevent OCR. Since preoperative high-frequency power and, in consequence, high vagal activity is less likely to be associated with significant OCR, some preoperative measures that increase vagal activity such as slowing the respiratory rate and increasing the tidal volume [12] may prevent the OCR.

Conclusion

We found that, in the OCR positive patients, preoperative time-domain measures, high-frequency power, and some nonlinear indexes of the RR intervals were significantly smaller when compared to those of the OCR negative patients. When these measures are applied to the patients who will get eye surgery, there is little chance to miss the patients who will eventually show significant bradycardia during the surgery.

Reference

[1]   Duncalf D, Gartner S, Carol B (1970), Mortality in association with ophthalmic surgery. American Journal of Ophthalmology 69: 610-5.

[2]   Mallinson FB, Coombes SK (1960), A hazard of anesthesia in ophthalmic surgery. Lancet 1: 574-5.

[3]   Lanza GA, Guido V, Galeazzi MM, et al (1998), Prognostic role of heart rate variability in patients with a recent acute myocardial infarction. American Journal of Cardiology 82(11): 1323-8.

[4]   Ho KK, Moody GB, Peng CK, et al (1997), Predicting survival in heart failure case and control subjects by use of fully automated methods for deriving nonlinear and conventional indices of heart rate dynamics. Circulation 96(3): 842-8.

[5]   Akselrod S, Gordon D, Ubel FA, et al (1981), Power spectrum analysis of heart rate fluctuation: A quantitative probe of beat to beat cardiovascular control. Science 213: 220-222.

[6]   Pincus SM, Goldberger AL (1994), Physiological time-series analysis: what does regularity quantify? American Journal of Physiology 266: H1643-56.

[7]   Bassingthwaight JB, Raymond GM (1995), Evaluation of the dispersional analysis method for fractal time series. Annals of Biomedical Engineering 23: 491-505.

[8]   Kanters JK, Hojgaard MV, Agner E, et al (1996), Short- and long-term variations in non-linear dynamics of heart rate variability. Cardiovascular Reseach 31(3): 400-9.

[9]   Arnold RW, Gould AE, MacKenzie R, et al (1994), Lack of global vagal propensity in patients with oculocardiac reflex. Ophthalmology 101: 1347-52.

[10] Kanters JK, Hojgaard MV, Agner E, Holstein-Rathlou NH. (1997), Influence of forced respiration on nonlinear dynamics in heart rate variability. American Journal of Physiology 272: R1149-54.

[11] Pincus SM, Singer BH (1996), Randomness and degrees of irregularity. Proceeding National Academic Science USA 93: 2083-2088.

[12] Eckerberg DL (1983), Human sinus arrhythmia as an index of vagal cardiac outflow. Journal of Applied Physiolology 54: 961-966.

Nonlinear Analysis of EEG in Patients with Schizophrenia

Jeong-Ho Chae, M.D.[1], Dai-Jin Kim, M.D.[1], Jaeseung Jeong, Ph.D.[2], Soo Yong Kim, Ph.D.[3], Won-Myong Bahk, M.D.[1], Tae-Youn Jun, M.D.[1], Kwang-Soo Kim, M.D.[1]

[1]Department of Psychiatry, College of Medicine, The Catholic University of Korea,
#62 Yoido-dong, Youngdeunpo-gu, Seoul, 150-713, Korea
E-mail: alberto@cmc.cuk.ac.kr
[2]Department of Radiology, Yale University, New Haven, CT, U.S.A.
[3]Department of Physics, Korea Advanced Institute of Science and Technology, Taejon, Korea

Abstract

We studied the complexity of the electroencephalogram (EEG) in schizophrenic patients by estimating the correlation dimension (D2) and the first Lyapunov exponent (L1) which might serve as an indicator of the specific brain function in schizophrenia. We recorded the EEG from schizophrenic patients fulfilling DSM-IV criteria and healthy controls at 16 electrodes, different from previous studies that recorded the EEGs at limited electrodes. We employed a method with an optimal embedding dimension to calculate the D2 and L1s. For limited noisy data, this algorithm was strikingly faster and more accurate than previous ones. Our results showed that the schizophrenic patients had lower values of the D2 and L1 at the left inferior frontal and anterior temporal regions compared with normal controls. These results for nonlinear analysis have some differences from those for power ratios in linear analysis. These suggest that the nonlinear analysis of the EEGs such as the estimation of the D2 and L1 might be a useful tool in analyzing EEG data to explore the neurodynamics of the brains of schizophrenic patients.

Keywords: Schizophrenia, EEG, Nonlinear analysis, Correlation dimension, Lyapunov exponent, complexity

Introduction

A detailed understanding of the neuronal mechanism associated with schizophrenia remains an elusive goal, not only because the human brain is an extremely complex organ, but also because it is relatively inaccessible to investigate during life.   Even though there are limitations in the spatial resolution, EEG may provide various information for neuronal activities of the live brain.   However it is generally agreed that no pathognomonic or characteristic EEG pattern is apparent on visual inspection [1].   A number of studies for schizophrenia with power spectral analysis over the last 20 years revealed that there are no definite specific findings for EEG in the patients [2].   This may be caused by the forcing of such a heterogeneous disorder as schizophrenia into group statistics, neglecting different subtypes and activities of a schizophrenic process.   One relatively stable finding is the increased beta power of the EEG in schizophrenic patients [3]. The increased beta activity may appear significantly on the left side, especially on the frontal area [2].   Gattaz et al [4] showed increased activities for fast alpha and beta bands in first onset, neuroleptic naive schizophrenics. The other consistent finding is the increased slow-wave abnormalities on the left side, especially on the left anterior temporal area and occasionally on the left frontal and parietal areas [5].   However, there are many limitations in using such a linear method because of the absence of an identified metric that quantifies complex behavior of the brain.

Recent progress in the theory of nonlinear dynamics has provided new methods for the study of time-series data from human brain activities [6].   Babloyantz et al [7] first reported that the EEG data from the human brain had chaotic attractors for sleep stages II and IV. Much research with nonlinear methods revealed that the EEG is generated by a deterministic neural process [8, 9].   According to these reports, the EEG has a finite non-integer correlation dimension and a positive Lyapunov exponent. Furthermore, the distinct states of brain activity had different chaotic dynamics quantified by nonlinear invariant measures such as correlation dimensions and Lyapunov exponents [10-12].   To the contrary, there is some evidence that the EEG is not a chaotic signal of low dimension [13, 14].   The above studies showed that the normal resting human EEG was nonlinear but did not represent low-dimensional chaos, and it may be generated from 1/f-like linear stochastic systems.

Regardless of what the true dynamics of the EEG are, nonlinear analysis of the EEG to make the correlation dimension and/or the first positive Lyapunov exponent estimates have proven to be very useful in making relative comparisons of different physiological states [15]. Many investigations with these methods have revealed possible medical applications for nonlinear analysis and have given rise to the possibility that nonlinear analysis of the EEG might be a useful tool in differentiating physiological and/or pathological brain states. In nonlinear analysis, some pathological conditions such as epileptic seizures, coma, and dementia showed an decreased complexity in EEGs, whereas normal attentional states tended to show an increased complexity by the estimation of the correlation dimension [16-22].

A few reported studies for the nonlinear analysis of the EEG of patients with schizophrenia have mainly estimated the nonlinear invariant measures of the EEG in schizophrenia at small numbers of electrodes [23]. We tried to record the EEG from 16 channels in patients with schizophrenia in order to investigate the whole pattern of chaotic dynamics in the brain by the D2 and L1.

Subjects and methods

1. Subjects

Subjects were 25 schizophrenic patients (12 men, 13 women; age = 25.1 ± 6.96 years, mean ± S.D.) fulfilling DSM-IV [24] criteria and 15 healthy controls (9 men, 6 women; age = 27.8 ± 4.24 years, mean ± S.D.). None of the patients had a history of neurological disorder and drug or alcohol abuse. The patients were taking neuroleptic medication with a mean dosage of 353.97 ± 319.36 mg chlorpromazine (CPZ) -equivalent. The fifteen control subjects were healthy individuals with no history of psychiatric or neurological disease who were selected from 30 volunteers. All subjects were right-handed and showed moderate amplitude (30-100µ V) on EEG background activities.

2. Data Acquisition and Analysis

The EEGs were recorded from the 16 scalp loci of the international 10-20 system. With the subjects in a relaxed state with closed eyes for 32.768 seconds, (16,384 data points) of data were recorded for the analysis and digitized by a 12-bit analog-digital converter in an IBM PC. Recordings were made under the eyes-closed condition in order to obtain as much stationary EEG data as possible. The sampling frequency was 500 Hz. Potentials from 16 channels (F_7, T_3, T_5, Fp_1, F_3, C_3, P_3, O_1, F_8, T_4, T_6, Fp_2, F_4, C_4, P_4, and O_2) against "linked earlobes" were amplified on a Nihon Kohden EEG-4421K using a time constant of 0.1 sec. All data were digitally filtered at 1-35 Hz in order to remove the residual EMG activity. Each EEG record was checked by inspection to be free from electrooculographic and movement artifacts and to contain minimal electromyographic activity. Whenever a decrease in vigilance was detected on the ongoing EEG, the technician instructed the patient to open his or her eyes, and a short pause was allowed, if needed, to minimize drowsiness. The recordings were obtained at approximately the same time of day (usually in the afternoon).

The first step in nonlinear analysis was to construct phase space using the delay coordinates proposed by Takens [25]. We used the time delays calculated by the method of mutual information to reconstruct the attractor. Time delays of 46-58 ms and embedding dimensions of 13-19 were used for the schizophrenic patients, and time delays of 28-32 ms and embedding dimensions of 11-19 were used for the control subjects. We used an algorithm of optimal embedding dimension, which was proposed by Kennel et al [26], to estimate the D2 and L1 fastly and efficiently for finite noisy data. This algorithm is strikingly faster and more accurate than other algorithms [27]. The values of D2 were calculated by using the Grassberger-Procaccia algorithm [28]. We calculate the first positive Lyapunov exponent L1 by applying a modified version of the Wolf algorithm [29]) and by following a proposal by Frank and colleagues [17]. A more detailed description of the procedure is presented in our previous paper [27].

Results

Patients with schizophrenia had lower average values of the D2 and L1 at the left inferior frontal (F_7: P= 0.005 and P= 0.012, respectively) and anterior temporal electrodes (T_3: P=0.016 and P= 0.025, respectively) compared with controls. The differences between the values of the D2 and L1 at the F_7 and T_3 channels are about 0.5 units. These results clearly show that for each subject, but different channels, the values of the D2 and L1 do not vary within broad ranges. To consider the antipsychotic drug effect, we estimated the relationship between CPZ equivalent dosage and the value of D2 and L1. There were no correlation between the D2 and neuroleptic dosage. Also the L1 does not correlate with the CPZ equivalent dosage except at O_2.

Conclusion

Nonlinear analysis has offered new tools for the investigation of information processing in the brain by analysis and classification of EEG signals. Compared with the conventional spectral analysis, nonlinear analysis of the EEG gives us information that reflects the dynamical properties of the whole brain.

There are some previous studies on complex behaviors of the schizophrenic brain. King et al [30] reported the results of the dynamic model of the central dopaminergic neuronal system, and suggested that the chaotic solutions of the dynamic equations correlate with the increased variability of behavior in schizophrenics. Koukkou et al [31] reported that the correlation dimension of the temporal-parietal EEG differed between the first-episode schizophrenics and normal controls. It was significantly higher in the first-episode schizophrenics than in normal controls. They suggested that the higher dimensional complexity of functional brain mechanisms in schizophrenics versus normal controls is reminiscent of the loosened organization of thought, and suggestions of certain superior abilities in the patients. Elbert et al [32] calculated the two descriptive measures (complexity and mobility) proposed by Hjorth and dimensional complexity of EEG from schizophrenic and normal subjects. They reported that most schizophrenic patients exhibited higher frontal dimensional complexity values than central ones. They suggested that the simultaneous perception of the delusional and the real world by schizophrenic patients might be based on increased information-processing power expressed through a higher correlation dimension, in which the frontal and the central dynamics are dissociated even in the relaxed waking state.

Our study is very different from previous studies with nonlinear analysis of the EEG in schizophrenic patients. First, we estimated the D2 and L1 in 16 channels, while previous studies estimated the correlation dimension in limited channels. Rőschke and Aldenhoff [23] reported a statistically significant decrease of the correlation dimension during sleep stage II and rapid eye movement (REM) sleep in schizophrenic patients. Rőschke et al [33] also reported that the L1 in REM sleep was significantly increased in schizophrenic patients compared with control subjects. From these findings, they suggested that correlation dimension meant complexity and the Lyapunov exponent implied flexibility; therefore, their results pointed to altered nonlinear brain dynamics during REM sleep in schizophrenia. In our results, we showed that schizophrenic patients in waking states had lower values of the D2 and L1 at the left inferior frontal and anterior temporal electrodes compared with controls. It means that the complex activities at the left fronto-temporal area in a schizophrenic brain are lower than those in normal ones. We cannot simply say that our results are opposite to the findings of Rőschke and Aldenhoff [23], since we measured the EEG of schizophrenic patients during waking relaxed states whereas they measured it during sleep states.

Our results of decreased complex activities at the left fronto-temporal area in the schizophrenic brain are consistent with the findings of hypofrontality and hypotemporality reported in the EEG studies on schizophrenia, cerebal blood flow, magnetic resonance imaging, and positron emission tomography. Recent investigation using power spectral analyses further suggested slowing of brain electrical activities, especially in the frontal regions. Several lines of evidence, including our results, imply that the frontal cortical area is dysfunctional in schizophrenia. Abnormalities affecting this region seem to be linked to impaired motivation, asocialization, and complex problems. The temporal lobe is also implicated in schizophrenia. Works on magnetoencephalography (MEG) in schizophrenia are consistent with left-sided disturbance originating from the superior temporal gyrus region. Our study revealed the hypofrontality and hypotemporality of the complexity in the left hemisphere of schizophrenic brains. It can be explained that the schizophrenic brain has decreased information processing and less flexible neural networks in the left fronto-temporal area.

One limitation was the fact that the antipsychotic medication was uncontrolled. In our study, the patients were taking neuroleptic medication with a range of CPZ equivalent doses, 0-1000 mg. Although there was no correlation between the nonlinear measures and neuroleptic dosages in most electrodes. our findings cannot be taken as proof of the absence of drug effects.

In conclusion, our results showed that schizophrenic patients in waking states had lower values of D2 and L1 at the left inferior frontal and anterior temporal electrodes compared with controls. These results suggested that the left frontal and temporal areas should be key to the etiology of schizophrenia due to lowered information processing. The implications of complexity in the brain may shed light on our understanding of the brain and its pathological condition. Especially, nonlinear measures of the electrophysiological activities in the brain may offer unique and fruitful perspectives for understanding important features of patients with schizophrenia.

References

[1] Itil TM (1977), Qualitative and quantitative EEG findings in schizophrenia. Schizophrenia Bull 3:61-79

[2] Hughes JR (1996), A review of the usefulness of the standard EEG in psychiatry. Clin Electroencephalogr 27:35-39

[3] Itil TM, Saletu B, Davis S (1972), EEG findings in chronic schizophrenics based on digital computer period analysis and analog power spectra. Biol Psychiatry 5:1-13

[4] Gattaz WF et al (1992), Hypofrontality on Topographic EEG in schizophrenia. Euro Arch Psychiatr Clin Neurosci 241:328-332

[5] Karson C et al (1988), Computerized EEG in schizophrenia. Schizophrenia Bull 14:193-197

[6] Pradhan N, Dutt DN (1993), A nonlinear perspective in understanding the neurodynamics of EEG. Computer Biol Med 23:425-442

[7] Babloyantz A, Salazar JM, Nicolis C (1985), Evidence of chaotic dynamics of brain activity during the sleep cycle. Physics Lett 111A:152-156

[8] Babloyantz A (1988), Chaotic dynamics in brain activity: In: Basar E (ed), Dynamics of Sensory and Cognitive Processing by the Brain. Springer, New York, pp. 196-202

[9] Soong ACK, Stuart CIJM (1989), Evidence of chaotic dynamics underlying the human alpha-rhythm electroencephalogram. Biol Cybernet 62:55-62

[10] Babloyantz A, Destexhe A (1987), The Creutzfeldt-Jacob disease in the hierarchy of chaotic attractor. In: Markus M, Muller S, Nicolis G (ed), Chemical to Biological Organization. Springer, New York, pp. 307-316

[11] Pijn JP et al (1991), Chaos or noise in EEG signals: Dependence on state and brain site. Electroencephalogr Clin Neurophysiol 79:371-381

[12] Röschke J, Aldenhoff J (1991), The dimensionality of the human's electroencephalogram during sleep. Biol Cybernet 64:307-313

[13] Osborne AR, Provenzale A (1989), Finite correlation dimension for stochastic systems with power-law spectra. Physica D 35:357-381

[14] Theiler J, Rapp P (1996), Re-examination of the evidence for low-dimensional, nonstructure in the human electroencephalogram. Electroencephalogr Clin Neurophysiol 98:213-222

[15] Rapp PE (1993), Chaos in the neurosciences: Cautionary tales from the frontier. Biologist 40:89-94

[16] Babloyantz A, Destexhe A (1986), Low-dimensional chaos in an instance of epilepsy. Proc National Acad Sci USA 83:3513-3517

[17] Frank GW et al (1990), Chaotic time series analysis of epileptic seizures. Physica D 46:427-438

[18] Pritchard WS, Duke DW, Coburn KL (1991), Altered EEG dynamical responsivity associated with normal aging and probable Alzheimer's disease. Dementia 2:102-105

[19] Pritchard WS et al (1993), Altered EEG dynamical responsivity associated with Alzheimer's disease: replication and extension. In: Jansen BH, Brandt ME (ed), Proceedings of the Second Annual Conference on Nonlinear Dynamical Analysis of the EEG. World Scientific, Singapore, pp.165-168

[20] Prichard WS et al (1994), EEG-based-neural-net predictive classification of Alzheimer's disease versus control subjects is augmented by non-linear EEG measures. Electroencephalogr Clin Neurophysiol 91:118-130

[21] Stam CJ et al (1995), Investigation of EEG nonlinearity in dementia and Parkinson's disease. Electroencephalogr Clin Neurophysiol 95:309-317

[22] Lehnertz K, Elger C (1998), Can epileptic seizures be predicted? Evidence from nonlinear time series analysis of brain electrical activity. Physica Rev Lett 80:5019-5022

[23] Röschke J, Aldenhoff J (1993), Estimation of the dimensionality of sleep-EEG data in schizophrenics. Euro Arch Psychiat Clin Neurosci 242:191-196

[24] American Psychiatric Association (1994), Diagnostic and Statistical Manual of Mental Disorders, 4th ed (DSM-IV). American Psychiatric Press, Washington DC

[25] Takens F (1981), Detecting strange attractors in turbulence in dynamical systems and turbulence. Lecture Notes in Mathematics 898. Springer, Berlin, pp.366-381

[26] Kennel MB, Brown R, Abarbanel HDI (1992), Determining embedding dimension for phase-space reconstruction using a geometrical construction. Physica Rev A 45:403-411

[27] Jeong J, Kim SY, Han SH (1998), Nonlinear dynamical analysis of the EEG in Alzheimer's disease with optimal embedding dimension. Electroencephalogr Clin Neurophysiol 106:220-228

[28] Grassberger P, Procaccia I (1983), Measuring the strangeness of strange attractors. Physica D 9:189-208.

[29] Wolf A et al (1985), Deterministic Lyapunov exponents from a time series. Physica D 16:285-317

[30] King R, Barchas JD, Huberman BA (1984), Chaotic behavior in dopamine neurodynamics. Proc National Acad Sci USA 81:1244-1277

[31] Koukkou M et al (1993), Dimensional complexity of EEG brain mechanisms in untreated schizophrenia. Biol Psychiatry 33:397-407

[32] Elbert T et al (1992), Physical aspects of the EEG in schizophrenics. Biol Psychiatry 32:595-606

[33] Röschke J, Fell J, Beckman P (1993),. The calculation of the first positive Lyapunov exponent in sleep EEG data. Electroencephalogr Clin Neurophysiol 86:348-352

Healing technology with Artficial Life

-Unconscious Flow-

Naoko Tosa

ATR Media Integration & Communication Research Laboratories
2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0288 Japan

tosa@mic.atr.co.jp

Abstract

In face-to-face communication, the occasional need for intentional lies is something with which everyone can identify. For example, when we get mad, circumstances may force us to put on a big smile instead of expressing our anger; when we feel miserable, good manners may dictate that we greet others warmly. In short, to abide by social norms, we consciously lie. On the other hand, if we consider the signs that our bodies express as communication (body language), we can say that the body does not lie even while the mind does. Considering this phenomenon, we propose a means of "touching the heart" in a somewhat Japanese way by measuring the heartbeat of the "honest" body and using other technologies to reveal a new code of non-verbal communication from a hidden dimension in society. We call this "techno-healing art."
Keywords: *A-life art,interactive art, emotion model,healing technology*

1. Introduction

The author believes that interactive art is one type of components that provides sympathy with communications. Interactive art can be thought of as one type of emotion and sympathy interface. It is familiar to us, and it forms agents or characters that can handle sensitive communications. In addition, they work on our mental states and emotional expressions, and on our character and intelligence, means that a person can also self-create his or her own personality. On the other hand, emotion recognition technology recognize only surface emotion of people [1][2][3]. I am interested in how to recognize unconsciousness feeling using computer-based interaction. Art is familiar to describe human unconscious emotion. I tried to realize in interactive art with technologies and techniques of art.
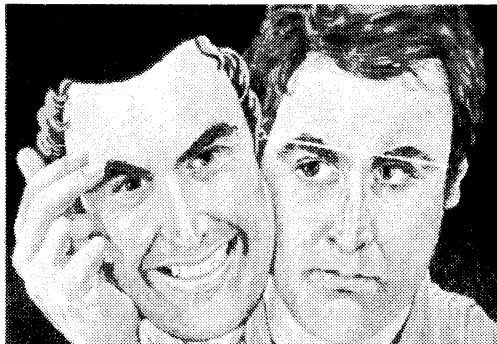
Fig.1 Mask of face

2. Concept of Unconscious Flow

Two computer-generated mermaids function as individual agents for two viewers. Each mermaid agent moves in sync with the heart rate detected by an electrode attached to the collarbone of its viewer. Then, using a synchronization interaction model that calculates the mutual heart rate on a personal computer, the two mermaids express hidden non-verbal communication. The data of relax-strain calculated from the heart rate and the interest calculated from the variation in the heart rate are mapped on the model. The synchronization interaction model reveals the communication codes in the hidden dimension that do not appear in our superficial communication. Then, using a camera to pick up hand gestures and a personal computer to analyze the images, the synchronization interaction model is applied to determine the mermaid's behavior. For a high degree of synchronism, the agents mimic the hand gestures of their subjects, but for a low degree of synchronism, the agents run away. As for background sound, the heart sound of the subjects are picked up by an heart rate sensor and processed for output on a personal computer for bio feed back. This work collaborate with Sony-Kihara Research Center, Inc.

Fig.2 In the bucket of [Unconscious Flow]

3.System

For installation, a space with four meters wide, four meters deep and three meters high is required. A dark and quiet space is preferable. Interactive actions are displayed on one main screen and two Japanese "shoji" screens. A

Japanese "hinoki" wooden bucket with a diameter of one meter that is filled with water is placed in the center of the installation,. Two persons, fitted with a stethoscope, experience non-verbal communication by touching their CG embodiments in the bucket. The synchronity based on the heart rate from the electrodes of the electrocardiograph is calculated by the PC, and the PC generates an arbitrary feeling in a CG form. The hand movements of the two persons are caught by an installed camera and an image analysis for the data is performed. In accordance with the synchronity interaction model, the CG embodiment either follows the movement of the hand of the partner with high synchronity or goes away from the hand of the partner with low synchronity. When one touches the CG embodiment of the partner, a vibrator gives him a simulated feeling of touch. The hrartrate sensor measures the timing of the heart, which is processed by the PC and outputted.

We were showing SIGGRAPH'99 Art Show.

Many nationality people intearct with [Unconscious Flow].
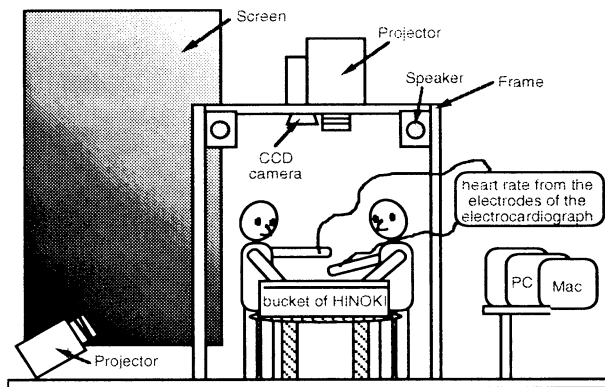
Fig.3-1 System of [Unconsious Flow]
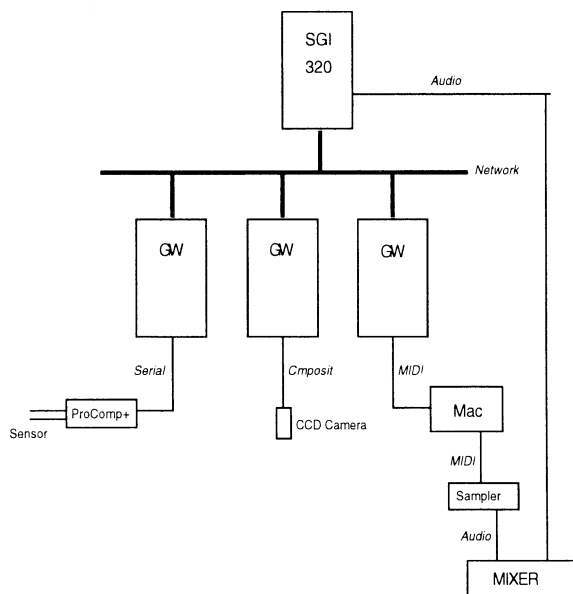
Fig.3-2 Hardware configuration
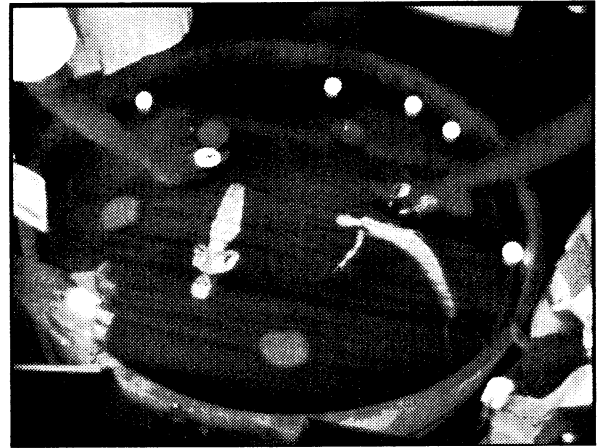
Fig.3-3 [Unconsious Flow] showing in SIGGRAPH'99

4. Synchronization interaction model

The data of relax-strain calculated from the heart rate and the interest calculated from the variation of the heart rate are mapped on the model. The synchronity interaction model reveals the communication codes in the hidden dimension that do not appear in our superficial communication.(Fig.4)

For example, (1) When both people are in the domain where they are highly relaxed and interested, they are considered synchronized. An animation is generated in which, for example, their CG-reactive embodiments join hands in brotherhood or enjoy friendly actions.(Fig.5)
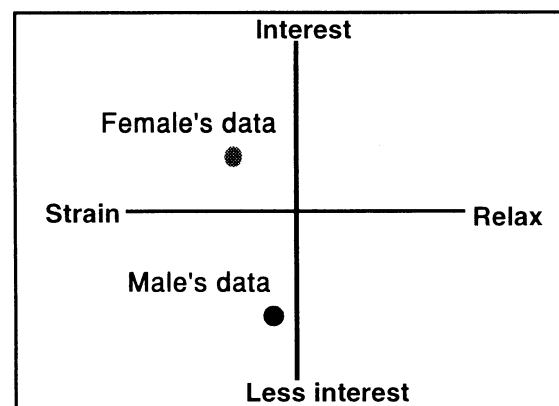
Fig. 4 Synchronity interaction model
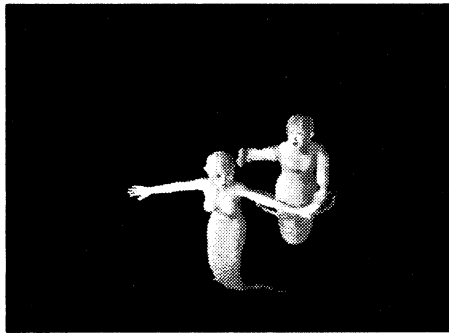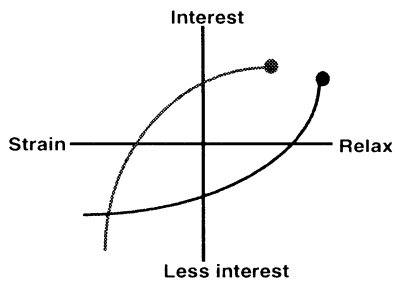
Fig. 5 Highly relaxed and Interested
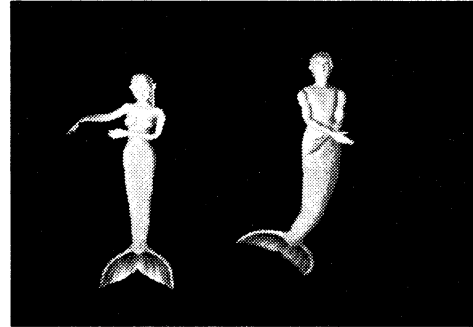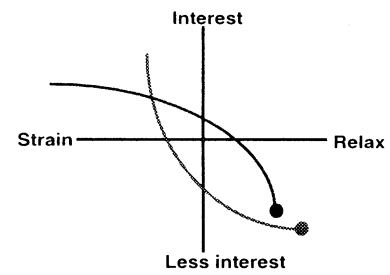
Fig.7 Highly relaxed and less interested

(2) When both people are in a situation where they are highly strained and less interested, unfriendly communication is generated. An animation is generated in which, for example, their CG embodiments quarrel with each other.(Fig.6)

(4)when two persons are in a situation where they are highly strained and highly interested, they are assumed to have stress and feelings of shyness. An animation is generated in which, for example, their CG-reactive embodiments behave shyly. In this way, (Fig.8)
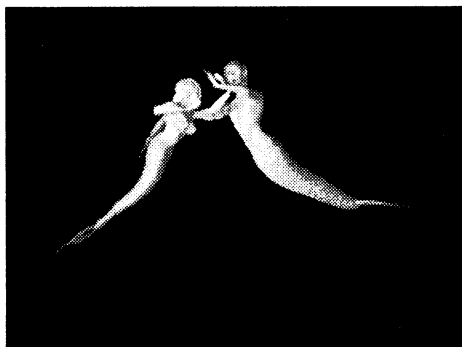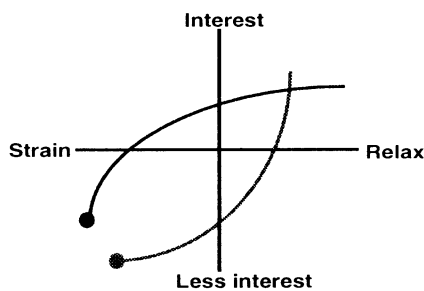
Fig.6 Highly strained and less interested
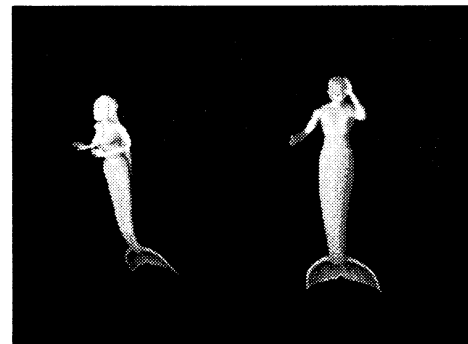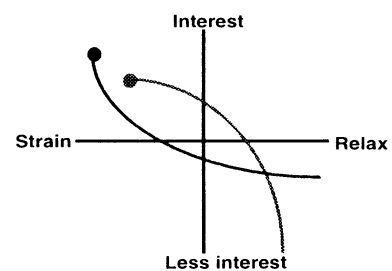
Fig.8 Highly strained and Highly interested

(3) When both people are in the domain where they are highly relaxed and less interested, they are considered, "going their own ways". An animation is generated in which, for example, their CG embodiments do not interfere with each other.(Fig.7)

new codes of non-verbal communication that can't be seen in face-to-face communication are found through the CG of the embodiments

5. Hand recognition

A person's hand with a marker is recognized by using CCD camera. CCD camera recognizes the position of two markers. The related program (Image Recognition) prosecces the distance between two markers and whether a hand touches a mermaid or not.
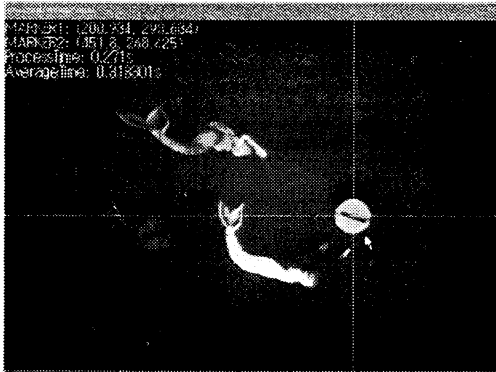
Fig.9 Image processing by hand recognition

6. Heart rate sensor

A person's heart rate is measured by putting the sensor on his finger. The heart rate is sent to a PC connected the heart rate sensor(ProComp+) via RS232C and is mapped on the synchronity model depending on the heart rate.

Fig.10 People put on heart reate sensor

7. Software configuration

Heart rate Analyzer is used to analyze the input data and to send the data to Event Control as event data. Event Control sends heart rate as MIDI commands to MAX/MSP program on Macintosh and some commands to Computer Graphics Generator if some Computer Graphics needs to be changed depending on the heart rete. Computer Graphics Generator creates Computer Graphics based on these commands and outputs the Computer Graphics. MAX/MSP program processes the sound data and the heart rate sound as required and then outputs it. Image Recognition analyzes the image data fed from a CCD camera and the relational information of the hand, and the Computer Graphics displayed is sent to Event Control and Computer Graphics Generator. Event Control sends some commands to Computer Graphics Generator if some Computer Graphics needs to be changed depending on the data.
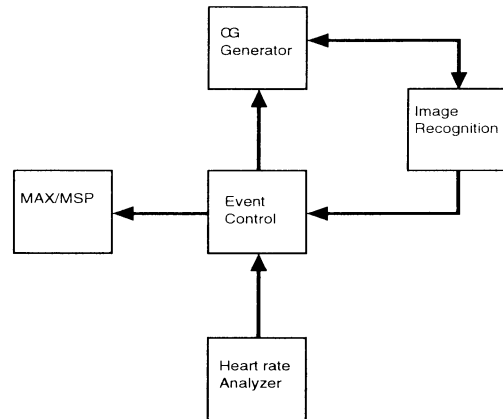
Fig.11 Software configuration

8. Conclusion

This work was exhibited at SIGGRAPH'99 that was held in Los Angeles. So many people visited the exhibition site and enjoyed the interaction of this Unconscious Flow. In the west coast the idea of healing and meditation is familiar. That is the reason why this work has been accepted by so many people. So far, this work uses biofeedback function based on heart rate of oneself. The other areas related to this work is sound hearing and hearing psychology. Future study will collaborate with these areas.

Healing technology with A-life will become very important in the future for human to human and even human to computer interactions. In the present day, people are exposed to various kinds of stresses in their daily life. In human to computer interactions as well as in human to computer communications, people want to have relaxed and less stressful communications.

9. Acknowledgments

Reference

[1]Tosa, N., and Nakatsu, R. : Life-like Communication Agent - Emotion Sensing Character MIC and Feeling Session Character MUSE-. in Proceedings of the International Conference on Multi-media Computing and Systems, 12-19, 1996.
[2]Maes, P., et al.: The ALIVE system: Full-body Interaction with Autonomous Agents. in Proceedings of the Computer Animation 95Conference, 1995.
[3]Reilly, S., :Building Emotional Characters for Interactive Drama, in proceedings of the Twelfth National Conference on Artificial Intelligence, Sattle, WA, 1994.

Future Communications Based on KANSEI Technologies

Ryohei Nakatsu

ATR Media Integration & Communications Research Laboratories

2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0288 Japan

E-mail:nakatsu@mic.atr.co.jp

1. INTRODUCTION

Against the conventional means of communications for which the main medium has been the telephone, it is necessary to ask whether or not a new means of communications should be sought that is suitable for the present multimedia era. Clearly showing this is the explosive spread of the Internet. Therefore, in what direction should the communications of the future be headed for? To learn the answer, we believe that it is necessary to understand global trends and to consider communications by returning to the basics.

From such a viewpoint, here, we first survey trends for the various media encompassing communications. Then, we explain the causes producing such trends by a model of communications. In addition, based on the above considerations, we consider the essence of communications that the communications technologies of the future should handle. Finally, we describe examples of research progressing at ATR Media Integration & Communications Research Laboratories and consider the relationships of these examples with ways of communications.

2. TRENDS OF COMMUNICATION MEDIA

We proposed "hyper-communications" as a new communications concept [1]. This was first based on the viewpoint that the boundary of communications media and other media would become ambiguous in the multimedia era and network era. Actually, such a movement is being caused by the various media (including communications). For example, a place for a new means of communications, the Internet, is being caused by the world of communications. It is thought that the Internet is a huge cyberspace that joins the whole world. People communicate with other people in that space and also shop.

In addition, looking at the movie industry, recent movies have been introducing digital techniques and computer graphics techniques and have been moving over to movies of a new generation. These techniques have given the ability to create very realistic worlds, i.e., cyberspaces, in which expression had been difficult in conventional movies. Also, video games, especially role playing games (RPGs), have made it possible for people to enjoy a story by becoming the main characters in a cyberspace. From these trends, it can be predicted for "communications in a cyberspace" to become one keyword for the new communications (Fig. 1).

Next, our attention is to focus on the information exchanged in communications. Figure 2 shows a model of communica-
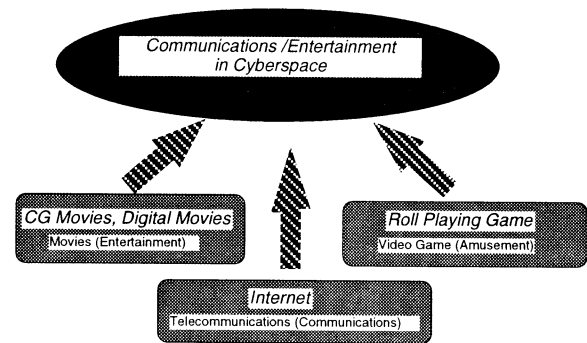
Fig. 1 New Trends in Various Kinds of Media.

tions for humans [2]. In the surface layer, a layer exists that manages communications based on the use of languages. It is possible to say that research on communications and information processing has come to deal with the mechanisms in this layer. For example, objects that have come to be treated in speech recognition are the logic information included in sounds.

The logic information, however, is only a part of the information composing the sounds. Other rich information contained in the sounds include emotion information and "kansei" information. It is considered that such information is generated by a layer in a deeper level, i.e., the interaction layer in Fig. 2 or the reaction layer.

The interaction layer manages acts such as the changes in the speaking turns, for the rhythm and production of the utterances, to maintain the communication channels. This layer plays an important role for smoothly carrying out communications for humans. Below the interaction layer is the reaction layer. This layer manages the more basic operations of humans. For example, this layer has functions such as to turn the face to the location from which a sound had come or to close the eyes at
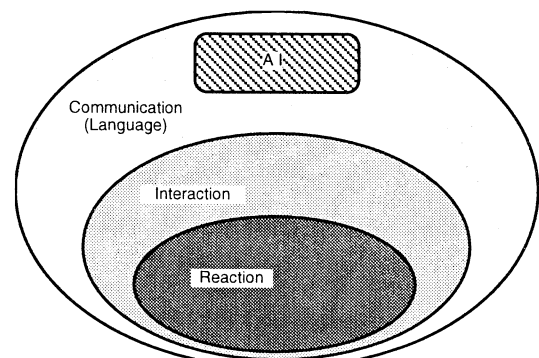
Fig. 2 Communication Model.

the sudden appearance of light.

In this way, it can be considered that human communications plays a role with important functions of a deeper layer (in addition to logical acts and the treatment of information), and that non-logical types of information like emotions and "kansei" are generated and solved through the movements of the above layers.

Conventional communications have come to apply logical communications, but in the future, other types of communications will become important. This, for instance, can be understood by observing the interest taken by young people in talking with portable telephones. In this case, it is possible to say that this phenomenon involves confirming sense-specific and emotions-specific relationships with partners and not the exchange of information with logical meanings [3]. Consequently, the transmission of non-logical information will become another keyword of the new type of communications.

From the above considerations, communications to transmit non-logical information by using a cyberspace can be expected to be the form of the new means of communications. For our part, we are progressing with research on concepts and prototypes that will concretely achieve this.

3. TARGETS OF COMMUNICATIONS IN THE FUTURE

To progress with research on such a new means of communications, it is important to consider how the ideal type of communications should be, by further going forward with the above findings. Logical information is sent from a sending side to a receiver. In the following phase, the sender and the receiver alternate their roles, and information is sent in the opposite direction. It cannot help but be said that such shear non-style communications is indeed the exchange of static information.

It can be considered that the original form of communications was more dynamic. Therefore, can the sender and the receiver, and the environment surrounding them become united, and is it not possible to understand the information as something shared between the sender and receiver, and moreover, as the relationship of the sender and receiver, and not as something handled as a physical amount separated from them?

From such a viewpoint, it can be thought that the elements the ideal communications should have include the following three items.

(1) Experience and synesthesia
(2) The integration of experience involving the body and mental experience
(3) The ability of active immersion

Below, explanations are given for each of the three items.

Experience and synesthesia

Communications to date have meant the sharing of information (related to communications) between people, by having logical information be sent and received. This, however, has indeed been "dry" communications.

The communications between a mother and her child, for instance, is close communications - both see and hear the same things in the same environment, and moreover, share the same feelings. Concerning this, more than to say the sharing of information, is it possible for one to call this a condition whereby a partner's feelings and one's own feelings are synchronized by having both experience the same experiences? Consequently, we want to point out "experience and synesthesia" as being the state of the original communications.

Concerning the term "experience", in addition to the meaning of "obtaining and sharing the same information", is not the implication strong for it to mean "to get information by moving the body"? Then, it should be possible to consider "experience = information + body". Next, concerning the term "synesthesia", it is possible to consider it as "strong thoughts about one's partner who has the same ideas and experiences". At this time, it can be considered that the "resonance phenomenon of some seeds" occurs between both sides. Consequently, it can be said that "synesthesia = sharing + resonance".

In other words, it is possible to say that the condition under which mutual thoughts are sympathized by the sharing of the same experiences, i.e., "experience and synesthesia", involves shallow communications; this is against the thinking of the conventional shallow communications (that is, to merely share logical information).

The integration of experience involving the body and mental experience

Next, we consider the contents of "experience". "Experience" is created from experience involving the body and mental experience. The former merely means to move the body. A representative example is exercise, but even a thing moving only a part of the body is experience involving the body. For example, experience involving the body includes the emitting of sounds. Moreover, even the action of viewing is an experience involving the body when attempting to view an external world because of the accompaniment of operations to move the head to the desired viewing direction, and so on.

In the past, only those operations involving aggressive body movements were called experiences having to do with the body, but here, we want to call actions having relationships to an external world by the use of the five senses including the actions of viewing and listening (which have come to be considered passive), as experiences involving the body.

For the latter (i.e., mental experience), it can be considered that the functions of the cerebrum become involved in actions added to the functions of the five senses such as using words, reading books, listening to music, and looking at pictures. Of course, here the handling of logical information occupies a large space, but the handling of "kansei" information such as emotions and feelings also occupies a large space.

It is in mental experiences when treating "kansei" information that we have an interest here. It is possible to say that all of the following are mental experiences of a high degree: expressing ones feelings by using words, becoming moved by reading a book, being impressed by listening to music, and becoming immersed in the world of a movie while watching it. The original meaning of the word "experience" involves an "integrated experience" merging the above experience involving

the body and mental experience.

The act of creating art like drama, performance, and sculpture has a side including experience involving the body - in that it achieves form by a practitioner being able to appreciate a high mental experience and moving the body with it. From this, it can be said that one is able to appreciate an "integrated experience" (combining experience involving the body and mental experience) when involved in the activity of the creation of such art. As a result of such an integrated experience, a kind of fascinating situation to appreciate is obtained during the art creation. Considering the above in this way, it becomes possible to explain why the media "karaoke" has become a popular media.

First, "karaoke" is an experience involving the body from the point of producing sounds. Moreover, it is also a mental experience from the point of reading and understanding words, i.e., the lyrics of songs, appealing to the emotions and feelings of humans. Consequently, it is possible to say that "karaoke" is a media to make one appreciate an integrated experience. Is there not a sense of fascination felt when singing "karaoke" that connects with and fits the sense of fascination an artist feels (said to be of a low level)?

In addition, as is generally said in many conversations, the mental aspect becomes more and more important for a sport when one enters the professional domain. This applies to baseball, among other sports. Therefore, if a sport reaches the professional domain for one by crossing over from the level of "fun", it is possible for that person to reach an integrated experience (i.e., experience involving the body and mental experience).

It is said that sports players often feel a sense of uplifting and a sense of fascination, but are these not the same as the sense of fascination felt when singing "karaoke" or the sense of uplifting an artist feels when creating art work? As explained above, an integrated experience is an experience called a "kansei" of a high dimension, and it is desirable to be able to achieve such an experience in communications among human beings.

The ability of active immersion

In an integrated experience, things able to be felt raise the sense of uplifting and the sense of fascination, but will anything be okay if they take the trouble of showing these? From this point of view, the character "Pocket Monster", which has recently become a topic of conversation, has become a kind of fascination and uplifting. There probably are various kinds of fascination and uplifting, but here, the word "immersion" is used as a general term for these. When looking at the Pocket Monster incident (i.e., many children being affected by seizures), it can be understood that there are many types of immersion. Here, we consider them by classification into "passive immersion" and "active immersion".

Does not the level of consciousness differ between both, although there is no change in becoming absorbed in both passive immersion and active immersion? Is there not a big difference for the condition of fascination, that is, the condition of losing consciousness by being mesmerized by viewing Pocket Monster? Consequently, the condition where one completely

forgets oneself or the condition where one loses consciousness (fascination, hypnosis, trance) becomes a key word that explains the feeling of passive immersion.

Active immersion, in contrast, makes possible the condition whereby one's consciousness is maintained in a normal manner when that person becomes absorbed. Conditions of immersion while clearly maintaining consciousness include the condition of doing work with concentration and the condition of immersion in the activity of creating art. Even in the case of sports, such conditions are experienced.

Do not these conditions involve expressing the condition of immersion for a professional baseball player or a professional golfer participating in a game, assuming "the nerves are able to concentrate on only the ball in front", "the ball thrown by the pitcher seems to have stopped", or so on? In this case, the consciousness is extremely clear and the actions of the person are able to be controlled completely. Unnecessary idle thoughts, however, are excluded.

When such passive immersion or active immersion pays attention to the processes that take place, an interesting fact becomes clear. It is the existence or non-existence of interaction.

Passive immersion does not work on a partner only by the receiving of information, like with Pocket Monster and mesmerism. In other words, there is a lack of interaction here. In contrast, active immersion differs from the point of working on the object, like with art creation and with sports. In other words, interaction exists with active immersion. Consequently, it is possible to say that the existence or non-existence of interaction is the key that distinguishes between passive immersion and active immersion.

Therefore, if interaction exists, will this result in active immersion? Unfortunately, the situation is not that simple. For example, let us consider the case of interactive art. Interactive art is said to be art where the picture or sculpture itself changes by movements, such as by the body and hand movements of the audience and their voices [3].

Novelties that interactive art asserts in comparison with conventional art are the achievement of active immersion by the intervention of interaction and the possibility of a higher appreciation for art than before. However, there is also a lot of interactive art that pulls the interest of the audience at the start but soon loses the interest of the audience, without leading to the achievement of active immersion. Is not a large reason for this because interest leans only towards the introduction of interaction and there is no deep insight into whether or not one aims at achieving something through the introduction of such interaction?

The objective of our research is the development of technologies and systems able to achieve an ideal form of communications. Next, we introduce an example of a concrete communications system which we are researching and want to consider how the characteristics of the communications described above can be achieved.

4. INTERACTIVE KALEIDOSCOPE[4]

Outline

Ever since the kaleidoscope was developed by D. Brewster in 1816, the interest from the reflections of the simple mirrors creating various shapes, has continued to attract people, and its popularity has persisted till today as a toy, among other reasons.

Iamascope is an electronic interactive kaleidoscope, and is a new type of kaleidoscope that uses image processing and computer graphics technologies. While being an interactive art work, it is also a musical instrument that integrates graphics of a new type (i.e., dance installment). Even if there are no techniques in the system to draw pictures and play musical instruments, the purpose of the system is to offer spaces in which the creation of multimedia content is possible; here, oneself and the audience consist of pictures and sounds.

The characteristic of Iamascope that largely differs from that of the conventional kaleidoscope is the fact that the user (i.e., a player) can, while penetrating as a material through the kaleidoscope, view and enjoy the pictures of the kaleidoscope. Here, it is possible to offer a location providing an aesthetic experience that is rich to the user and the audience by the addition of music.

Contents of Iamascope

Figure 3 shows a block diagram of Iamascope. A large-size projector-type screen is used for display purposes, and below it, a video camera is set. The output of the camera is distributed by two graphics workstations (GWSs); one is responsible for kaleidoscope generation and the other for music production.

The video output of the GWS responsible for kaleidoscope generation is displayed in the vicinity of the user through the projector. The serial output of the GWS responsible for music production is connected to a MIDI music synthesizer and involves a system where music can be heard through round speakers from the source.

With the kaleidoscope generation sub-system, video images are transferred to texture memory, and from there, a suitable domain is used and kaleidoscope images are created. For example, because it is possible to obtain reflected images by the

mirrors of the kaleidoscope when the field image element fragments are inverted along a certain edge, by repeatedly using the obtained patterns, the kaleidoscope pattern of two mirrors is completed when a polygon approximation circle is made. This processing is possible at approximately video rate when using the technique of video texture mapping (implemented by hardware).

Furthermore, even the music subsystem for music generation uses the movements in the domain of the same image element fragments. The music subsystem takes the time change in brightness for each picture element from the input image fragments and generates commands that sound the MIDI source according to the changes. Consequently, with the brightness changes caused by the movements of the body, and so on, in front of the camera, it is possible for music to be played.

By making the input for the display of the kaleidoscope and the control of the music generation the same image fragments, these tasks are basically done by the same system. For the user or player, the linking (i.e., correspondence) of the changes of the kaleidoscope and the performance of the music becomes possible, and the sight and hearing feedback becomes a part of the control system. Figure 4 shows a state using Iamascope.

Evaluation

Let us evaluate Iamascope from the point of the characteristic for which the ideal communications system discussed in 3 should prepare.
(1) Experience and synesthesia

Iamascope is an experience-type system. Experience here, i.e., creating beautiful images and music by using one's body, is made possible even for regular people. Furthermore, by having several people share the experience together, these people are able to jointly experience creating artistic pictures and sounds unable to be experienced in daily life, and strong synesthesia is produced among them.
(2) The integration of experience involving the body and mental experience

Iamascope allows one to experience an integrated experience. Experience involving the body, to move one's body for expression purposes, becomes basic. Furthermore, it resultantly be-
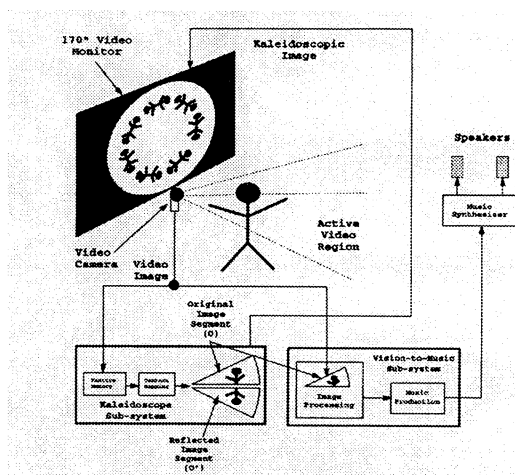
Fig. 3 System Configuration of "Iamascope."
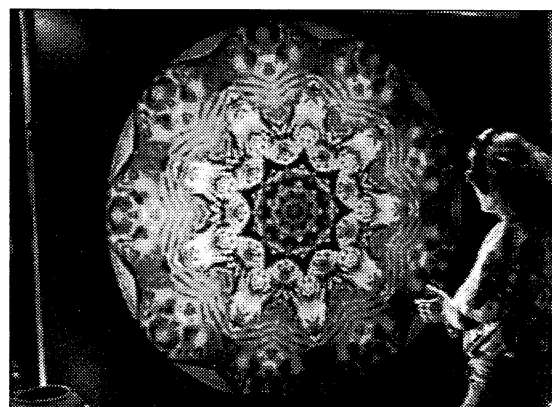
Fig. 4 Interaction with "Iamascope."

comes possible to obtain a mental experience, to appreciate occurring beautiful kaleidoscope pictures and music. Because the change of a beautiful form connects with the generation of a more beautiful picture, the experience involving the body and the mental experience are deeply connected. Consequently, one is made to meet with the integrated experience.

(3) The ability of active immersion

Iamascope is an "active" system with its meaning being to generate pictures and music by one's own movements. Moreover, the corresponding interaction becomes a simple composition that is in real time and where movements are reflected in the kaleidoscope picture generation just as they are. As a result, at the same time simple pictures and music are created with simple movements, when the movements are advanced to the movements of a professional dancer, there is the characteristic that the pictures and music also are advanced. In other words, when a professional dancer performs according to the movements, background kaleidoscope pictures and music are generated. From this, the user or player becomes the system and has an easy time experiencing the feeling of immersion.

5. INTERACTIVE POEM [5]

Outline

An interactive poem is a poem of a new type whose world is completed jointly while a human being and a computer read out the phrases of the poem mutually. In a usual poem, in contrast, the poet tries to express his own feelings and "kansei" by connecting carefully chosen words, phrases, and sentences. With the power of words held by these words and phrases, people are able to understand the message(s) the poet is attempting to express and sink into the world of the poem made by the poet. The world of a poem created by a poet, however, can be called static, because the words, phrases, and sentences employed there and their order are fixed.

The concept of the proposed interactive poem tries to cross the limit which the usual poem has, by taking in an interaction function. By having a computer agent mutually read out the phrases of a short poem with a participant, it is possible to complete the world of a new poem that merges the world of the poem of the computer agent and the world of the poem of the participant.

A computer agent called "Muse" appears on the screen. The face of Muse is designed beforehand to a form suited towards
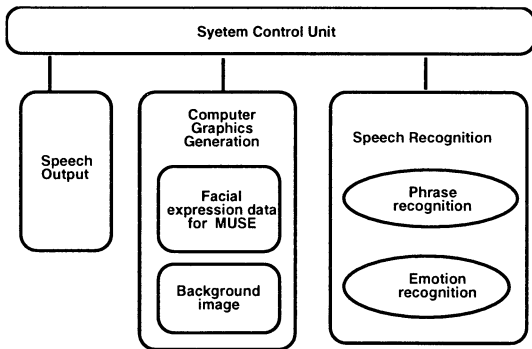
Fig. 5 Software Configuration of "Interactive Poem."
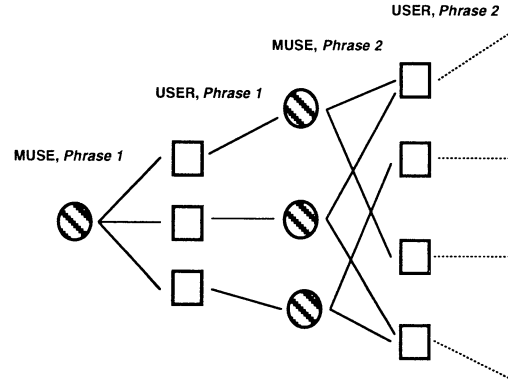
Fig. 6 Phrase Transition for "Interactive Poem."

expressing the world of a poem. Muse reads out the phrases of the short poem. By listening to them, the participant chooses phrases of the poem so as to answer to Muse from among the multiple phrases displayed on the screen, and talks to Muse. Through such an interaction, Muse becomes a cooperative poet and completes the new poem and the world of the poem with the participant.

Interactive poem system

Figure 5 shows the software construction of the system. The control unit stores the database of the interactive poem, and controls the operation of the whole system using this. The interactive poem database has a network construction like that shown in Fig. 5. A typical poem is a continuation of phrases. That the computer agent and the participant (i.e., human) read it out alternately, is the simplest case, but the world of the poem becomes simple.

Here, as shown in Fig. 6, multiple phrases are prepared that can connect with the phrases read out by the computer agent and the user chooses them. By doing things in this way, it is possible to make the world of the created poem have diversity.

The speech recognition unit has a phrase recognition unit, which recognizes the phrases read out by the user, and an emotion recognition unit, which recognizes the emotions contained in the voice of the user. The phrase recognition part has an unspecified speaker speech recognition function based on a Hidden Markov Model (HMM). In addition, the emotion recognition part has an emotion recognition function based on a neural network. The kinds of emotions able to be recognized are eight in number, as follows: "happiness, joy, anger, fear, amiableness, ridicule, disappointment, and neutral".

The reactions of the computer agent "Muse" to the sounds of the user are expressed in her voice and appearances. The expressions on the face of Muse are determined in reaction to the emotion recognition results. The types of phrases that Muse should read out next and their voiced emotions are determined from the phrase recognition results and the emotion recognition results.

A professional radio announcer utters the phrases (beforehand) that Muse should read out in the eight types of emotions, they are stored in the audio output part, and are output following the instructions of the control unit. In addition, animations

Fig. 7 Interaction with "Interactive Poem."

of the eight types of emotions of Muse are created (before-hand), are memorized in the image output part, and are output following the instructions of the control unit. At the same time, various background pictures suitable for the contents of poems are memorized in the image output part, and are output by the instructions of the control unit according to the progress of a poem. Figure 7 shows a state of actual interaction.

Evaluation

Even for the interactive poem, like with Iamascope, an attempt is made to perform an evaluation from the viewpoint of the base element of the communications described in 3.

(1) Experience and synesthesia

Recently, the reading out of poems has been re-evaluated, and quite a few conferences for the reading out of poems have opened. Poems were, originally, more read out than read.

With interactive poems, by the reading out of a poem, it becomes possible to participate in a collective experience to create a poem jointly with a computer poet. In addition, from this, it is possible for the emotions and "kansei" the original poem wants to transmit to be better understood. In other words, it is considered that synesthesia is borne towards the poet who made the poem. As a result, experience and synesthesia are expressed.

(2) The integration of experience involving the body and mental experience

The reading out of poems has a place that resembles the above-mentioned "karaoke". In other words, at the same time the experience involving the body occurs in reading out a poem, it is possible to appreciate a mental experience, i.e., to appreciate the art of advance words (i.e., the poem) and to understand it. That is, it is possible to appreciate an integrated experience (which integrates the experience involving the body and the mental experience). Moreover, it can be considered that the poem will try to resort to more advance feelings, whereas most "karaoke" texts try to directly resort to the emotions of humans.

(3) The ability of active immersion

An interactive poem can provide an "active" experience, in the meaning of "to complete a poem by interaction using words with a computer agent". This can be understood if it is actually experienced; it is easy for one to enter the condition of some type of immersion when proceeding to read out a poem through the power of words held by the poem. With that meaning, the

proposed system can be considered to have the power to achieve active immersion to some degree.

6. CONCLUSION

It can be considered that the communications technology of the future must expand applicability to communications relating non-logical information, such as "kansei", from conventional communications which assumes the sending and receiving of logical information. It is possible to say that the spread of the current new communications represented by the Internet and so on supports such a movement.

Here, we considered the characteristics the ideal communications, including the communication of "kansei" information, and so on, should prepare for. In addition, we introduced a part of our research aiming to achieve such communications. Our research involves studies obtained from projects by engineers equipped with the characteristics of artists and from artists working in cooperation with engineers.

We are aggressively promoting a trial involving the above which we call "art & technology" [2]. The system explained here was found to satisfy, to some degree, the condition of each element to make the ideal communications materialize. Therefore, it can be considered that with the communication systems in the multimedia era, it might become difficult to judge whether or not there is acceptance by the user, because of the phenomena of the fusing of communication media and other media, the fusing of technology and content, and so on. In such cases, it can be considered that the above "art & technology" proposes one piece of effective methodology. In the future, we hope to promote research based on this methodology.

7. REFERENCES

[1] R. Nakatsu, "Toward the Creation of a New Medium for the Multimedia Era," Proceedings of the IEEE, Vol. 86, No. 5, pp.825-836 (1998).

[2] R. Nakatsu, "Image/Speech Processing that Adopts an Artistic Approach -Integrating Art and Technology-," Proceedings of ICASSP1997, pp.207-210 (1997).

[3] S. Weits, "Nonverbal Communications," New York: Harcourt, Brace, Jovnovich (1974).

[4] S. Fels, D. Reiners and K. Mase, " Iamascope: An Interactive Kaleidoscope," Visual Proceedings of SIGGRAPH97, pp. 76-77 (1997).

[5] N. Tosa and R. Nakatsu, "Interactive Poem," Proceedings of AIMI International Workshop, pp. 54-59 (1997).

The Artist and The Automaton:

Evolving musics, a second nature, artists gently break the news...

Rodney Berry
ATR Media Integration and Communications
Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288,
Japan.
rodney@mic.atr.co.jp
www.mic.atr.co.jp/~rodney

Keywords: artificial life, music, art, virtual worlds

Abstract: I am an artist and musician working with artificial life in the context of exploring the boundaries between the animate and inanimate. Two works are briefly described, *Feeping Creatures*, an alife music work, and the *Vital Presence* project at ATR Media Integration and Communications Laboratories, Kyoto. From a personal perspective, I believe that artists can have a beneficial role in shaping society's emotional response to a potentially unsettling period of change. Self-designing technologies will greatly affect human beings' relationship to technology and nature. This could be confronting for some but the advantages may outweigh the disadvantages on a social level. Meanwhile, technological artists will enjoy new delights as their works come a-live.

I am an artist and musician with an interest in new technologies both for their functionality and for their possible meanings. Over the last decade or so, I have produced sound works and music in a variety of media. In almost all these works, there has been a common thread concerning itself with the boundary between animate and inanimate. The two most recent projects have been specifically focused on artificial life. The first of these is called *Feeping Creatures.*

Feeping Creatures is an interactive virtual environment. The software was developed in collaboration with three programmers[1] from a company called Proximity in Sydney Australia. Its inhabitants have sequences of musical notes encoded in their chromosomes. The creatures, called *feeps*, mate according to the relative amount of musical consonance or dissonance between them and their immediate neighbours. Some prefer partners to be more consonant where others prefer more dissonance. Their offspring inherit a mixture of each parent's note series to make a new melodic fragment. These fragments are repeated over and over as a *feep* wanders around its flat little world. Rhythm is roughly analogous to energy or food in their world. Rhythmic elements in the form of *duration* and *articulation* values are obtained from the *trees* that the *feeps* eat.

The user enters a dark room and sees the world as a flat green grid projected on the wall. *Feeps* are shown as coloured cubes moving around the grid and *trees* are shown as green triangular pyramids. The user can move his or her point of view (a virtual camera) around the world by moving a mouse. Spatialised audio comes from four speakers around the room allowing the user to hear the *feeps'* songs all around. Sound is placed according to a *feep's* location on the grid giving strong audio cues as to where they all are in relation to the user. The user can only look and listen at this stage and must be content with being an invisible tourist. Later, I hope to introduce predators, more interaction between the user and the environment and more sophisticated genetics but, for now, the programmers are too busy making a living to develop the piece further.

The second a-life music project has been given an overall working title of *Vital Presence*, referring to my continued attempts to create in an artefact, the sense of being in the presence of a living thing. The project is being developed at ATR's Media Integration and Communications Laboratories in Kyoto, Japan. Collaborating with me are two student intern researchers, Joy Nicholson and Gavin Wong. In practise, the environment so far looks a little like that of *Feeping Creatures*, although the underlying mechanisms are different. The creatures of this world, called *spinners* have a long chromosome made of a string of binary digits. Traits are arbitrarily mapped to segments of this string. We have focussed on evolving timbres (spectral content or *colour* of sound) and rhythms. Melody, at the time of this writing is part of the food chain (similar to rhythm in *Feeping Creatures*). Spinners start life as a *soprano* with a high voice before they reach puberty and become an *alto*. *Altos* initiate mating with the older *tenors* and *basses*, later stages in the life cycle. An *alto*, a *tenor* and a *bass* must be present for conception to take place. The offspring inherits part of each of the three parents' chromosomes and whatever traits come with it. The shape of the creatures' body is also genetically determined and I hope to evolve other functional aspects of the creatures' existence as well. The physics of the world will be governed by musical factors along with the biology. Eventually, it will be possible for a musician to play an instrument and have the software make new creatures containing those notes and

rhythmic elements. They may also be able to influence the timbres of new creatures. The resulting music will be a result of the system, its inhabitants and the user responding to each other and adapting organically.

Tools such as this represent what I see as an imminent change in the way people use tools and technology. Our tools have always been modelled on products of nature. The first tools were probably actual parts of animals, teeth and shells for cutting and scraping, skins to insulate and protect the body. I imagine that later we abstracted the functional principles of these things and applied them to other materials. As we began to work with metals and use the power of combustion, our artefacts grew to resemble less and less the forms of nature. Early steam engines replaced many manual labourers but also required many operators to stop the machines from getting out of control. Because of this, ways were quickly found to make the machines self-regulating and thus more economically viable. Kevin Kelly[2] writes that the industrial revolution contained a hidden information revolution. By placing the centrifugal governor on his steam engine, Watt gave his machine enough primitive self-awareness to get information about, and adjust its own internal states. Such use of feedback is an essential quality of living organisms, so it is not surprising that biological metaphors were often applied to such machinery. Even today, we use the word *plant* to describe industrial installations.

The industrial revolution brought many natural forces under human control. It was assumed that the mysteries of nature would quickly fall before the technologically enhanced gaze of modern man. To a large extent, this has come to pass. As the underlying processes of nature became more clearly understood, we begin to manipulate even the stuff of life itself. However, along with answers and solutions, our amplified senses also reveal to us fresh questions and problems. The more we rely on the mediation of machines to look at the world, the more we are tempted to describe what we see in mechanical terms. Despite all this, nature is still full of surprises and its processes are more complex and far less rational and mechanistic than we once wanted to believe. The process of evolution manages to be simultaneously efficient but also spectacularly redundant. By combining favourable traits from ancestor organisms, natural selection uncannily produces body types that are ideally suited to their surrounding environment. As long as it works, it survives, even if some traits make no difference whatsoever. In the long run, over large populations of individuals, this *shotgun approach* works well.

For all its redundancies and haphazardness, this kind of *irrational design*[3] in nature is applied to technology. For more than ten years now, we have used genetic algorithms to evolve optimal solutions to given software problems. The solutions are often weird, but they can work as well as those written by human programmers. The comparatively recent use of *universal logic arrays*, *FPGAs* etc. to produce evolvable electronic hardware indicates another milepost along the same trail. I see this as the beginning of a new journey for humans as toolmakers. We can only speculate as to where this trail leads or how far along it we can go. One thing we can be sure of however is this: When we look at any device or machine in current use, even if we do not know how it was made or how it works, we know that somewhere, there is somebody who does. In the very near future, this will no longer be the case. Soon there will be products on our supermarket shelves, in our stores, in our homes, even inside our bodies, the precise workings of which will be unknown even to our finest engineers.

I think that the growing popular realisation of this fact will be as significant as the discovery that the Earth is not the centre of the universe. It will be as confronting as finding out that human beings share common ancestors with the apes and all other living things on this planet. The sudden awareness that we are no longer masters of our technology will be an enormous blow to our collective ego. In fact, biologically inspired technology could become a kind of second nature, just as mysterious as the organic nature it emulates.

Many people will find this change challenging or even frightening but personally, I do not see this to be such a bad thing. I think that biological paradigms could influence technology to work more in harmony with existing biological nature. As the technosphere and the biosphere merge into a new integrated entity (*a bionisphere?*), humans will need to adopt a new relationship with their technology. The relationship will become more a sense of stewardship than ownership, an attitude appropriate for a sustainable technological future. I suspect· that this relationship should also feel somehow familiar to us. Whether from the distant past or the present, a farmer or a hunter-gatherer does not fully understand the workings of the animals he exploits for survival. A horse has inputs and outputs that the farmer needs to be aware of. He also needs to know how to care for it and keep it healthy. It is not necessary to be able to build a horse from scratch. He simply breeds for desirable characteristics and trains the horse for tasks that are in keeping with its basic functional structure and inclination.

A farmer's relationship with a horse is functional but mixed with affective and emotional elements. This allows a bond of co-operation and symbiosis between the farmer and the horse. Perhaps the appearance of Tamagotchi and virtual pets indicate our first tentative yearnings for this kind of relationship with our machines. Rather than being subject to an out-of-

control self-designing technology, maybe we can be partners.

Perhaps this utopian vision is achievable. Then again, maybe these concepts can only be applied as a form of *damage control* in the face of a horrible and alien future. Either way, I feel that artists have a strong and important role in both the development of these technologies. It is important to me that more artworks based on biological paradigms find their way into the attention of the public. I think it is an essential part of peoples' adjustment to a rapidly changing world. In a session concerned with healing, I believe that art can help allow ideas to reach the public in a way that allows them to approach them in a spirit of play. It provides us with a chance to get used to some new things; maybe to play in virtual reality is beneficial before having to go to work there. Play will be very important to help society incorporate such technological change into its conscious and subconscious fabric

For the artist, the search for new toys (I really mean tools) goes on, along with the search for new ways to interpret what happens in the world around us. A-life and generative art-forms offer the artist strong alternatives to prevalent multimedia practice. It will soon be easy to abandon pointing and clicking around a fixed and finite database in favour of something more fluid and organic. Instead of climbing around the branches of someone else's decision tree we will soon be roaming in our own symbolic forests. The artist can now step out of the tiresome role of God/Creator to become a hunter or farmer of aesthetic experience.

---

[1] Tom Mander, Ben Ross and Brian Murray

[2] Kelly, Kevin - *Out of Control* 1995 Fourth Estate, London; A&W paperback edition; ISBN: 1857023080 pp147-148

(also on the web: http://panushka.absolutvodka.com/kelly/5-0.html)

[3] ibid, p387

Integrating Real and Virtual Worlds in Shared Space

Ivan Poupyrev [1], Mark Billinghurst [2], Hirokazu Kato [3], Richard May [2]

[1] MIC Research Labs	[2] HIT Lab	[3] Faculty of Information Sciences
ATR International	University of Washington	Hiroshima City University
Hikaridai, Seika, Souraku-gun	Box 352142	3-4-1 Ozuka-Higashi, Asaminami-ku
Kyoto 619-02, Japan	Seattle, WA, 98195 USA	Hiroshima 731-3194, Japan
poup@mic.atr.co.jp	grof@hitl.washington.edu	kato@sys.im.hiroshima-cu.ac.jp

Abstract

In the Shared Space project, we explore, innovate, design and evaluate future computing environments that will radically enhance interaction between human and computers as well as interaction between humans mediated by computers. In particular, we investigate how augmented reality enhanced by physical and spatial 3D user interfaces can be used to develop effective face-to-face and remote collaborative computing environments. How will we interact in such collaborative spaces? How will we interact with each other? What new applications can be developed using this technology? These are the questions that we are trying to answer in research on Shared Space. This paper provides a short overview of Shared Space, its main directions, core technologies and applications.

Keywords: augmented reality, physical interaction, computer vision tracking, collaboration, entertainment.

1. Introduction

In the Shared Space project, we explore, innovate, design and evaluate future computing environments that will radically enhance interaction between human and computers as well as interaction between humans mediated by computers. In particular, we investigate how augmented reality enhanced by physical and spatial 3D user interfaces can be used to develop effective face-to-face and remote collaborative computing environments.

Shared Space integrates a number of novel interface technologies, including:

Augmented reality. Augmented reality (AR), i.e. overlaying of virtual objects on the real world, allows us to integrate computer-generated and computer-controlled objects into surrounding us everyday physical reality [6]. Unlike virtual reality where the physical world is completely replaced with synthetic environments, in augmented reality environments, 3D computer graphics objects are mixed with physical objects to become in a sense an equal part of the real world.

Collaborative computing. Using computers can be a lonely experience: normally, there is no support for collaborative activities in which several people can work together. In the real world collaboration, objects and information can be simultaneously and asynchronously accessed by multiple participants at once, with communication discourse flowing freely between the participants. Shared Space aims to allow for a similar freedom of collaboration and impromptu level of inter-

action that we have in physical environments. We also aim to address some of the limitations of current collaborative interfaces. These limitations include the lack of spatial cues, the difficulty of interacting with shared 3D data, the introduction of artificial seams into a collaboration, and the need to be physically present at a computer to collaborate [8, 12].

Physical interfaces. Interaction in today's graphical user interfaces (GUIs) is often dubbed as *direct*, meaning that the user "picks" and "manipulates" interface objects using mouse similarly to how we actually pick and manipulate physical objects. When compared to early command line interfaces, interaction in current GUIs is indeed more direct; nevertheless, it can only be loosely compared to our interaction with the physical world. In fact, interface objects do not have physical properties, and "picking" and "manipulating" them are simply metaphors that help us understand how to use the interface by drawing from our everyday experiences. Shared Space investigates the use of *physical, tangible* interfaces [9] where instead of using a mouse and a keyboard the user can control computer by physically manipulating multiple simple physical objects that become a part of the user interface.

Spatial 3D user interfaces. 3D user interfaces, an important topic in virtual reality, explore how users can efficiently and effectively interact in spatial 3D computer-generated environments. In spatial interfaces as well as in the physical world, users are not constrained by the 2D metaphor of conventional desktop user interfaces but can interact freely in space. Shared Space is a 3D user interface that provides the user with rich spatial cues and combines spatial and physical interaction for easy control and manipulation of virtual objects.

Computer vision tracking and registration. Computer vision techniques have recently become very popular in user interface research [7] The core technologies employed in Shared Space – tracking and registering virtual objects in the physical world – are based on a heavy use of computer vision [2].

The rest of this paper is organized as follows. In the next section, we briefly discuss related work, followed by a more detailed discussion of the technologies involved in Shared Space: *augmentation, collaboration, interaction* and their implementation based on computer vision tracking and registration techniques. We then describe an application, which was demonstrated at SIGGRAPH99, that uses all of these technologies to build an entertainment application.

2. Related work

Shared Space has been inspired by a number of previous research projects in augmented reality and ubiquitous computing research, computer supported collaborative work (CSCW), 3D user interfaces and virtual reality, and tangible and physical computing [16]. Our research on Shared Space integrates many of these individual components into an effective interface that can support intuitive face to face 3D CSCW.

While the use of spatial cues and three-dimensional object manipulation are common in face to face communication, tools for three-dimensional CSCW are still rare. One approach is to add collaborative capabilities to existing desktop-based three-dimensional packages. However, a two-dimensional (2D) interface for three-dimensional collaboration can have severe limitations, for example users find it difficult to visualize depth cues or the different viewpoints of their collaborators [10].

Alternative techniques include using large stereo projection screens to project a three-dimensional virtual image into space, such as in the CAVE system [5]. Unfortunately, images can only be rendered from a single user's viewpoint in this setting, so only one person will see true stereo. While this might be satisfactory for some tasks, such as collaborative viewing, effective face to face CSCW using CAVE is impossible.

Mechanical devices can be used to create volumetric displays, such as scanning lasers onto a rotating helix to create a three-dimensional volumetric display [15]. However, these devices do not allow direct interaction with the images because of the rotating display surface.

Multi-user immersive virtual environments provide an extremely natural medium for three dimensional CSCW. Research on the DIVE project [4], GreenSpace [11] and other fully immersive multi-participant virtual environments has shown that collaborative work is indeed intuitive in such surroundings. Participants can seamlessly exchange and communicate gesture, voice and graphical information. However, most current multi-user VR systems are fully immersive, separating the user from the real world: notes, documents, tools and other artifacts of everyday life cannot be easily accessed from immersive virtual environments.

Unlike other methods for three-dimensional CSCW, augmented reality interfaces can overlay graphics and audio onto the real world. This allows for creation of AR interfaces that combine the advantages of virtual environments and possibilities for seamless interaction with real world objects and other collaborators.

Single user AR interfaces have been developed for computer aided instruction [6], medical visualization [1], information displays and other purposes. These applications have shown that AR interfaces can enable a person to interact with the real world in ways never before possible. However, although AR techniques have proven valuable in single user applications, there has been significantly less research on collaborative, multi-user applications. The AR2 Hockey [12] and the Studierstube project [14] are two of the few exceptions.
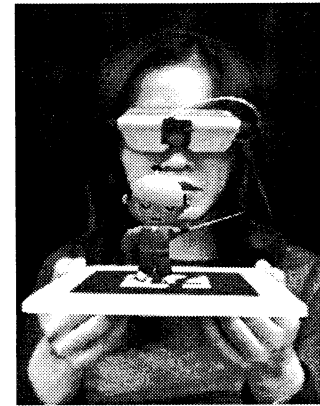
Figure 1: The HMD and a camera are used for registering and viewing virtual objects, e.g. a samurai model, that appear on top of the physical 2D markers.

On the interface side while the physical and tangible interfaces have been extensively explored [9], there have been few efforts at combining them with spatial interfaces.

Finally, computer vision techniques have been extensively used to track and register virtual objects in augmented reality applications. Our approach was inspired by the work of Rekimoto who developed a technique for robust tracking of 2D markers and registering 3D objects relative to them [13].

3. Shared Space

This section discusses key aspects of Shared Space, i.e., augmentation, collaboration, interaction, and implementation based on computer vision tracking and registration techniques.

3.1 Augmentation

Shared Space uses a see-through or opaque head-mounted display (HMD) with a lightweight camera mounted in front of the display (Figure 1). With the opaque display, the output from the camera is directed to the HMD so that the user has the illusion of looking through a display in the environment in front of him or her. The AR environment includes a number of marked objects with square fiducial patterns that have an identifying symbol in the middle of the pattern. When the user looks at the objects, computer vision techniques are then used to identify the specific marker, calculate head position and orientation relative to the fiducial marks, and display 3D virtual images so that they appear precisely registered with the physical objects. The details of the implementation are briefly described later in the paper, for a full description see [2].

3.2 Collaboration

Shared Space allows users to refer to physical notes, diagrams, books and other real objects while at the same time viewing and interacting with virtual images. More importantly, users can see each other's facial expressions, gestures and body language, thus increasing the communication bandwidth between the users. Furthermore, the Shared Space interface supports three-

Figure 2: In collaborative environment of the Shared Space, users can see and interact with physical and virtual objects, and also see the other participants.

Figure 3: Spatial interaction in Shared Space: users trigger animation of virtual objects (in this case alien and UFO) by bringing two cards together.

dimensional collaboration between face-to-face and remote users so that multiple users in the same or remote location can simultaneously work in both the real and virtual world (Figure 2). Since all users share the same database of virtual objects, they see the same virtual objects attached to the markers. The user can pick up and show the card to the other participants, pass or request virtual objects in the same manner that we do with physical objects in the real world.

3.3 Interaction

Shared Space explores the use of spatial and physical interaction in augmented environments: the user can directly manipulate virtual objects by manipulating marked physical objects with virtual objects "standing" on them (Figure 1). The system can robustly track the motion of the physical markers and keep the virtual object precisely aligned relative to the marker. Because

several markers are tracked simultaneously and the relative positions of marked objects to other marked objects can be calculated, we could also exploit spatial relations between augmented objects. In the example in Figure 3, the user can trigger the animation of virtual objects (e.g. UFO and alien) by placing cards next to each other.

3.4 Tracking and registration

Shared Space uses a computer vision based tracking and registration algorithm designed and implemented by Hirokazu Kato [2]. Figure 4 shows the basic scheme of the algorithm.

4. Applications and user experiences

A number of applications have been developed and explored using various components and configurations of the Shared Space technology, including mobile conferencing space for remote collaboration (AR Conference Space [3]) and a collaborative entertainment application, which was demonstrated at the Emerging Technologies exhibit at the SIGGRAPH99 conference. The goal of this demonstration was to show how augmented reality could be used to enhance face to face collaboration in a way that could be used by novices with no training.

A multi-player game similar to the game concentration was designed. We presented visitors with sixteen 5x7 inch playing cards with tracking patterns on one side, and the visitors were required to match cards. Three people could play simultaneously and when they turned the cards over they saw a different 3D virtual object on each card, such as a witch, horse, alien, or crab (Figure 3 and Figure 4). The goal of the game was to match objects that logically belonged together, such as an alien and a flying saucer. When cards containing logical matches were placed side by side, an animation was triggered involving the objects on the card. For example, when the card with the virtual witch on it was placed next to the card with a virtual broom on it, the witch would to jump on the broom and start to fly around in a circle. Sound cues have been also used to help players to identify correct matches: a unique musical sequence was played for each successful match. Since the players were all co-located they could easily see each other, and the virtual objects.
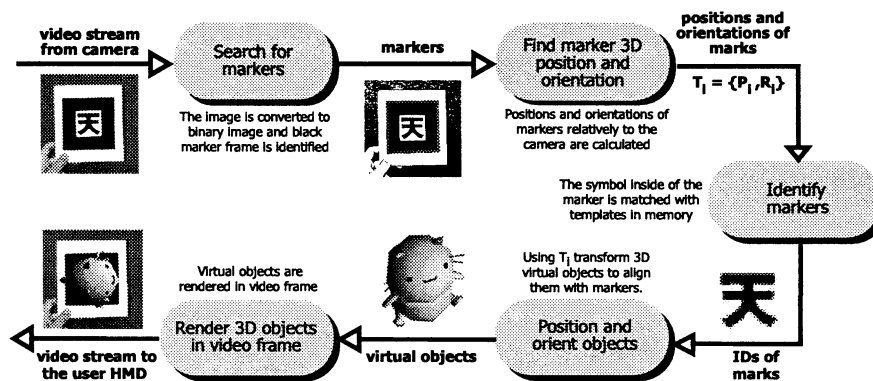
Figure 4: Shared Space tracking and registration technique.

Over the course of the week of August 7-13 at SIGGRAPH, around 3000 conference participants tried the exhibit. Users had no difficulty using the AR interfaces and exhibited the same sort of collaborative behavior seen in typical face to face interaction with physical objects. For instance, during the game players would often spontaneously collaborate with strangers who had a matching card, request and pass cards around as well as collaboratively view objects and completed animations. Furthermore, since the matches were not obvious, users would often request and receive help from other collaborators at the table.

The physical, tangible nature of our interface made collaborative interaction very easy and intuitive. Users passed cards between each other, picked up and viewed virtual objects from all angles and almost always expressed surprise and enjoyment when they got a match and the static virtual objects came to life. By combining a tangible, physical interface with 3D virtual imagery, we found that even young children could play and enjoy the game (Figure 5). Users did not need to learn any complicated computer interface or command sets – the only instructions people needed to be given to play the game were to turn the cards over, not cover the tracking patterns, and find objects that matched.

Users also commented on how much they liked the image recognition and on how little lag there was in the system. This comment is interesting because there was actually a significant (200-300ms) delay, however, the users became so immersed they did not notice this.

Figure 5: The Shared Space at the SIGGRAPH99

5. Conclusions

In our work on Shared Space, we combine real and virtual worlds to create compelling 3D collaborative experiences in which the technology transparently supports normal human behaviors. It is this transparency that is a key characteristic of the Shared Space research and should enable the continued development of innovative collaborative AR interfaces in the future. More information on the Shared Space can be found at: http://www.hitl.washington.edu/research/shared_space/ or http://www.mic.atr.co.jp/~poup/research/ar/

6. Acknowledgments

We are very thankful to ATR computer graphics artists Keiko Nakao and Jun Kurumisawa who designed and built models and animations for the SIGGRAPH Shared Space exhibition. We are also grateful to Shigeo Imura for his help in designing and implementing the computer graphics elements of the software platform. We are also thankful to all HITL and ATR researchers and management, especially Prof. Furness, Dr. Ohya and Dr. Nakatsu for their support of the demonstration as well as HITL students and researchers who prepared and worked the SIGGRAPH demonstration.

7. References

1. Bajura, M., Fuchs, H., Ohbuchi, R., Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery Within the Patient. *Proceedings of SIGGRAPH '92.* 1992. ACM. pp. 203-210.

2. Billinghurst, M., Kato, H., Collaborative Mixed Reality. *Proceedings of International Symposium on Mixed Reality (ISMR '99).* 1999. Springer Verlag. pp. 261-284.

3. Billinghurst, M., Kato, H., Real World Teleconferencing. *Proceedings of CHI'99, Extended Abstracts.* 1999. ACM. pp. 194-195.

4. Carlson, C., Hagsand, O., DIVE - A Platform for Multi-User Virtual Environments. *Computers and Graphics,* 1993. 17(6): pp. 663-669.

5. Cruz-Neira, C., Sandin, D., Defanti, T., Kentyon, R., Hart, J., The CAVE: Audio Visual Experience Automatic Virtual Environment. *Communications of the ACM,* 1992. 35(6): pp. 65.

6. Feiner, S., MacIntyre, B., Seligmann, D., Knowledge-Based Augmented Reality. *Communications of the ACM,* 1993. 36(7): pp. 53-62.

7. Freeman, W., Anderson, D., Beardsley, P., Dodge, C., Roth, M., *et al.*, Computer vision for interactive computer graphics. *IEEE Computer Graphics & Applications,* 1998. 18(3): pp. 42-53.

8. Ishii, H., Miyake, N., Toward an Open WorkSpace: Computer and Video Fusion Approach of TeamWorkstation. *Comm. of the ACM,* 1991. 34(12): pp. 37-50.

9. Ishii, H., Ullmer , B., Tangible bits towards seamless interfaces between people, bits and atoms. *Proceedings of CHI97.* 1997. ACM. pp. 234-241.

10. Li-Shu, Flowers, W., Teledesign: Groupware User Experiments in Three-Dimensional Computer Aided Design. *Collaborative Computing,* 1994. 1(1): pp. 1-14.

11. Mandeville, J., Davidson, J., Campbell, D., Dahl, A., Schwartz, P., *et al.*, A Shared Virtual Environ-ment for Architectural Design Review. *Proceedings of CVE '96 Workshop.* 1996.

12. Ohshima, T., Sato, K., Yamamoto, H., Tamura, H., AR2Hockey: A case study of collaborative augmented reality. *Proc. of VRAIS'98.* 1998. IEEE. pp. 268-295.

13. Rekimoto, J., Matrix: A Realtime Object Identification and Registration Method for Augmented Reality. *Proceedings of Asia Pacific Computer Human Interaction (APCHI'98).* 1988.

14. Schmalsteig, D., Fuhrmann, A., Szalavari, Z., Gervautz, M., Studierstube - An Environment for Collaboration in Augmented Reality. *Proceedings of CVE '96 Workshop.* 1996.

15. Soltan, P., Trias, J., Dahlke, W., Lasher, M., McDonald, M., Laser-Based 3D Volumetric Display System: Second Generation. In *Interactive Technology and the New Paradigm for Technology.* 1995 pp. 349-358.

16. Weiser, M., Some computer science issues in ubiquitous computing. *Comm of ACM,* 1993. 36(7): pp. 75-84.

Sensing, Communication and Intentionality in Artificial Life.

Stephen Jones
ATR - Multimedia Integration and Communication Laboratory
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-02 Japan
sjones@mic.atr.co.jp

Introduction

I want to examine several examples of Artificial Life (AL) systems in which sensing and therefore the requisite sensors are used as a functional part of the system, ie used with intentionality on the part of the system designer and with intentionality on the part of system "creatures".

It is one of the basic concepts of AL that it involves the simulation in computing systems, robots and other hardware/software configurations, of various functions which operate within, so to speak, "normal" biological life. These processes usually consist in the simulation of evolutionary processes and the use of genetic algorithms to develop independently behaving, or autonomous, entitities which are able to carry out some subset of the possible distinguishing characteristics and processes of living systems. These often include processes of sensing, feeding (or other survival maintenance), mating and reproduction, mobility and the like. The entities being considered are usually some kind of cellular automaton which has some of the characteristics which make up the idea of "agenthood".

In examining sensing in AL systems I want to explore the role of sensory processes and their extension into the areas of comunication between agents and the development of intentionality within an agent. The processes of sensing, communication and intentionality all contribute greatly to the question of the actual assessment of the success of AL projects in their implementation of aims such as the emulation and recapitulation of the evolution of biological organisms from bacteria to humans.

Ultimately, for the purposes of any autonomous "biological" agent it is the processing of information that is the primary motive in sensing the the organism's context and in the organism's engaging in communication. Definitions of my use of these terms are in order:

> **Information** is what is carried in a physically embodied (or simulated version thereof) difference relation generated by aspects of an organism or coherent entity within the context of some system. This concept of information is derived from Bateson [1] where, in the context or environment of the "organism", it is news of a difference and within the "living" system it is the difference which makes a difference. (For a discussion of information and its embodiement see Jones [2])

> **Sensing** amounts to a coherent system's (ie. an "organism's") capacity to absorb difference relations from its context and to carry out such transforms (processing) of those sense-data as to have them available as information about that context.

> **Communication** is the excretion (outputting) of a probe into the context in order to elicit a sensible

response from that context. When sensible to another entity, which may or may not respond, a communication between "creatures" can occur (is enabled).

Intentionality may be said to appear when the sensory and/or communicative act is produced in the "direction" of an object in the context for the specific purpose (intention) of eliciting information from the object.

Given that the intention of AL experimentation involves the emulation of biological processes I suggest that the concepts of an organism and a coherent system or autonomous agent have sufficient similarity for them to be reasonably interchangeable within this discussion. In fact, I am arguing that the issue of sensory and communcation systems and the entity's intentionality is a guiding criterion for establishing the success of, and understanding areas of failure in, artificial life projects.

Perhaps more strongly I would argue that the *use* of sensory and communicative processes by autonomous agents may be a more fundamental criterion than the generation of such agents or organisms by evolutionary processes, given that it is not yet proven that algorithmic top-down type processes *must* fail to produce AL organisms.

So having said that it is incumbent upon me to demonstrate what would be a sensory-cum-communicative process at the lowest possible level and to indicate where intentionality might become relevant. I will then look at some examples from a by-no-means-exhaustive review of the literature.

Sensory systems

For all practical purposes the single cell is the lowest level unit of coherent systems which demonstrates the basic attributes of a living system. A virus, though more minimal in level, fails to demonstrate at least one generally considered necessary aspect of living systems and that is independent rather than parasitic reproduction, in that it requires the agency of a host's genetic transcription services for its replication. (Of course what this implies about sexual reproduction I'd rather not go into here.)

In cellular behavior, whether as a single celled creature operating independently within some biochemically active environment or as one of myriad cells in some large scale organism, the process of being able to sample and sense the chemical soup in which the cell operates is a basic necessity. This is the only way it can acquire the basic chemical feedstocks for protein synthesis of any and all of the molecules its system requires for whatever processes it has evolved to exercise. So the activity of opening the cell membrane to transport a molecule from outside to inside

though essentially a transport function is in fact a primary sensory act. The shape of the membrane pore may well be such as to only recognise one particular molecule but this is of no consequence since in the high level of intelligent animals the sensory pore (or port) known as the eye is, itself, designed specifically to recognise only a very limited range of but one kind of physical particle, the photon of visible light.

If the cell, as is most likely, should excrete, via a membrane transport process, some kind of chemical (by)product of its internal processes and should some other cell find that that molecule has some kind of significance for it, that is, is recognised by some membrane pore, or is sensed by that other cell, then a communicative act has occurred, minimal as this may seem in this case. When looked at in the context of, say, neuronal cells, this process then becomes highly significant and the communicative function utterly clear. This is exactly synaptic transmission and it is what gives us all the capacity to understand and operate within the world, our chemical and cultural soup. (For a detailed discussion of the sensory and informational aspects of the biochemistry of synaptic transmission and nervous system activity in general see Black [3])

So it can be seen here that communication is a sensing process augmented by an output process which within the context produces two-way interaction. Now, at this level it is probably specious to suggest that this kind of communication demonstrates any form of intentionality, but I would suggest that autocatalytic interaction with the chemical environment of the cell and the mutual development of cells in general would have been a necessary part of the process of the evolution of pre-biotic systems, just as this bootstrapping is a necessary part of all biological systems.

Intentionality appears when the cell or agent (perhaps by now, necessarily a multicellar organism, though it is by no means proven that single cells do not show intentionality given some of the more interesting aspects of cell behaviour in being able to modify its chemistry to accommodate serious shifts in the content of its environmental soup) actively produces a probe into its environment in order to elicit some difference relation and potential information from some recognised aspect thereof. In other words the sensing now has to be for the "cell's" own autonomous purposes, whether it wants to know who's around the district or if it's good to eat. Autonomous behavior depends on intentionality even if this is highly stripped back.

Sensory Processes in AL

I shall now look at some examples of AL which profess to involve at least sensory processes.

Craig Reynold's Boids
Boids are a set of autonomously behaving virtual birds - boids (code objects) in a common contextual space (of computer memory) - which are aware of other similar code objects sharing the same context. Essentially each boid in the flock is a distributed processing element in the system. They are able to behave as a flock of apparently co-ordinated entities on the basis of three rules:
    1: avoidance of collisions with neighbours.

2: matching velocity (speed and direction) with neighbours
3: maintaining a local centering within the flock by staying close to neighbours.
These rules, operating for each boid in a local region, then allow a graded set of relationships throughout the larger global flock such that the autonomous behaviour of each boid provides an overall coherence to the flock.

The information that the boids need for navigating according to these rules is provided by a perceptual model which "make[s] available to the behaviour model approximately the same information that is available to a real animal as the *end result* of its perceptual and cognitive processes." (Reynolds [4], my italics.) This simulation does not, therefore, involve sensing in a way consistent with what we normally think of as sensing. In fact it bypasses any sensing at all and provides what amounts to the output of the sensing system directly to the behavior model in the boid. This output is a filtered selection of the available data in the context memory, providing position, orientation and velocity data with the sensitivity of data about the neighbors decreasing exponentially with distance from each particular boid.

Thus no actual sensing takes place but the equivalent result of sensing is provided directly to the brains of each boid so that they in effect have senses and their behavior is directly dependent on this "virtual" sensing substitute. It would not, in my opinion, be overly charitable, to say that this is perfectly good sensing within the abstract way that I am discussing the term. The boids have available to them the sense-data as information that the system provides at each time-step of its process, and so given the distributed "brains" of each boid in the context space, each cell of the system is sensing the difference relations available in its environment. I don't think, however that this legitimises some claim that communication is taking place, and I don't think that Reynolds is making such a claim at all. Further, since the sensing process is a programmer-imposed aspect of the boids behavioral routine occurring regularly with each time-step of the system's execution any claim of intentionality could be at best only borderline.

Werner and Dyer's communication experiments
Werner and Dyer used a toroidal (wrap-around) grid of 200 x 200 squares as an environment in which to evolve a male and a female version of the same genotype to explore the possibile evolution of communication between the two types. In this purposefully restricted situation the "animal's" task was to find mates. The females could only emit sounds in response to sighting a male within a close region around them and the males could only hear the signals and move on the basis of their interpretations of "instructions" contained in those signals. This way the environment exerted some pressure to evolve communication rather than having an external operater impose their view of how the communcation might work. As they put it "To put evolutionary pressure on the animals to communicate, we needeed to design animals in an environment such that some animals would have information that other animals needed to know but were not capable of finding out for themselves. The animals with this valuable information would have to communicate it to the other animals." (Werner and Dyer [5])

The signal generation and the interpretation of instructions were conducted by recurrent neural nets. When the female "sighted" a male the activation of the input layer flowed through to produce a signal at the output layer which could be interpreted by the male as an instruction to move in a particular direction, (forward, left or right). The male's task was to find the female for mating. The strategies developed - ie. the neural net weightings - are completely determined by the genome of each creature so that any signal produced or any interpretation of a signal is a function of the evolved genome of that creature. Modifications of strategies occurred through mutation and crossover in the sexual reproduction of the parental genomes.

Sensing is built in and communication is seen to evolve, but because the signalling and interpretations are deterministic this is communication without intentionality as Werner and Dyer point out. The situation is strictly behaviouristic, a stimulus (a result of a sensing) producing a fixed response which may or may not be appropriate to the mating task. The development of good strategies was effected by the simple process of removing aging genotypes given a fixed population size in the environment. If a mating could not take place then there would be no survival of that genome.

Terzopoulos, et alia: Artificial Fishes
In Terzopoulos, Rabie and Grzeszczuk's [6] artificial aquatic environment their Artificial Fishes "navigate purposefully through their world by analyzing the retinal image streams acquired by their eyes". These virtual eyes are able to focus on an object in the environment by foveation and controlling the direction of gaze, bringing an object from low resolution peripheral vision into the foveal centre for high resolution vision. Those objects which are focussed upon are externally designated "prey" fishes, other objects not so designated are ignored. Here the sensing is done to read the environment for a particular intentional act, namely hunting, but the intentionality is imposed from outside - as a set of preset fixed habits such as hunger, fear and mating - rather than acquired through some sort of evolutionary selection pressure.

Importantly the information produced by the sensing process has direct operational use or effect in the creature for the purpose of muscle control both in directing the gaze and in effecting motion control through the muscle actuation system. Terzopoulos *et al* also use something approaching internal or proprioceptive sensing to enable the Artificial Fishes to actually learn to swim in a productive way, say for hunting. This amounts to an iterative updating of parameters taken directly from muscle driver functions optimized against discrete objectives. Though much of the learning of specific muscle control algorithms is done in the Fish's "brain" Terzopoulos *et al* intervene in the process to promote controller objectives with certain externally desired characteristics thus guiding the optimization of the motion control algorithms.

Terzopoulos *et al* claim that the fishes navigate "purposefully" through their aquatic environment and purpose can readily be seen to have a level of intentionality therein, but it isn't easy to distinguish the degree of autonomous agency involved in their Artificial Fishes. Much of the algorithms used are built externally from various optimization functions and then imposed into the fish. For example they arbitrarily chose to use an approximation of the primate vision system in their fish's vision guidance system. But, of course, the evolution of autonomous agency isn't exactly their interest in developing this system.

Tom Ray: Tierra
Tierra is a system of digital organisms or cells (machine-code programs) which live in a purpose built environment provided by the Tierra Operating System. Networked Tierra is a version of Tierra in which the cells "are able to sense conditions on other machines and move between machines". As Ray points out the objective of the project "is not to create a digital model of organic life, but rather to use organic life as a model on which to base our better design of digital evolution." (Ray and Hart [7])

In Networked Tierra each of the organisms (machine-code cpu-structures) starts with a pair of differentiated "tissues" in the ancestor cell/dna. The two tissues provide a reproductive system and a sensory system. Over a sequence of reproductions the cells may lose or modify their tissues. The main purpose of the sensory tissue is to probe other Tierran ecologies on other networked machines to enable a mother cell to determine which of the probed Tierran environments is the best one to send its daughter cell. The sensory tissue sends a "tping" code (a Tierran ping command) out to 15 other networked machines and, by comparison of the data returned by the Tierran operating system in each machine, makes a decision as to the currently best environment to which to migrate the daughter cell. Examples of the kinds of criteria the mother cell uses for its determination are speed of system per number of cells in system and average reproduction rate.

The cells in this system clearly demonstrate intentionality in their sensory process but equally clearly they are completely unaware of any other cells in the system and communicate only with the environment. Thus there is no communcative process simply a sensory activity. One could not make claims towards the emulation of living processes here and as Ray makes quite clear his intention is not such emulation but simply the exploration of evolution as a means for developing programs.

Sommerer and Mignonneau: Pico_Scan
Pico_Scan is an interactive installation artwork built at ATR near Kyoto. The work is based on graphical elements each of which is an artificial organism in itself. These are displayed on video screens and the audience can view the results and interact with the process through the use of a purpose-built scanning device.

Genetic algorithms sense the environment through the scanner and data produced by the scanner is used by these routines as parameters for graphical displays which they generate. These displays are modified on the basis of distance of an object from the camera/scanner, overall colour of the scanned surface, absolute position in the physical environment and the volumetric body capacitance of the person holding the scanner. These data sources provide the graphic production routines with some sense of the audience in the installation space, their engagement with the work in using the scanners and where they are in that space. Thus the system senses what is going on around it and the audience for the work is integrated into the complex graphical productions of its virtual-life processes.

This is the only example I give of an AL system which is capable of sensing its physical context rather than simply the data or memory space in which it operates. To what extent there is internal sensing and communication between cells generating images I am not privvy, but I suspect the cells are quite independent, simply evolving in isolation according to the use of the scanners by outside agents. Of course Sommerer and Mignonneau's intention is to produce a pleasing visual experience in which the audience has a major role in the creative process. This the graphical results of *Pico_Scan* most certainly provide. (For a report of their other works see Sommerer and Mignonneau [8])

Discussion

If the work of the digital designer is to produce evolved digital entities which either resemble living systems or interact with living people, then sensory processing and some level of communicative activity is essential. Mignonneau and Sommerer's works, as interactive art works, are among few AL works which express any interest in the human. All their works show a desire for direct interaction with the audience for the work. (Sommerer and Mignonneau [8]). Other works which do elicit direct human interaction in the evolutionary process include Karl Sims' *Genetic Images* in which the audience has direct involvement through aesthetic judgement in the provision of selection pressure in the evolution of the images. (discussed in Ray [9])

Although the other works I have discussed here may produce interesting visual output, especially Reynolds' *Boids*, the primary motivation is in the further development of programming technologies and the visual output is largely artifactual. Thus the kinds of criteria I am drawing out are not operating in these works. These criteria, that the AL system should show some level of sensory and possibly communicative function as well as intentionality if there is to be any autonomy in the "organism", aim towards bringing the productions of AL evolutionary processes into contact with human processes and activities so that the systems do not grow in isolation from the human environment. To produce creatures which exist in some sort of isolation is in the long run to fail the creatures themselves and force them into a kind of confinement which we, under the same conditions, would consider cruel. In other words to become human requires the presence of other humans in the bootstrap into the cultural world and in the long run it would be bizarre to insist on complete isolation for any created entity. The kinds of entity that we might wish to keep at bay in this sense are produced under exactly those conditions of isolation in which *we* would become psychotic. Though the thought of this kind of future development might seem far fetched it is probably good prophylaxis to hold these matters in mind in the early stages of our attempts at artificial creation. If the enhancement and humanisation of communications and communication technologies is our goal then to do it under "de-humanised" conditions surely will breed failure.

But the real point about these kinds of criteria is that autonomy implies self-regulation, which is impossible without some sort of knowledge, not simply about the "organism's" internal processes, but also about the consequences of these processes on the environment in which the organism operates. In other words the organism has to sense its environment in order to gain information which may directly effect its continued existence. This is especially so as the organism, and others in the same context, will produce outputs which effect the environment and other entities in that environment, thus effecting fitness criteria and the fit of the organism within that context. To know of what you are doing to the environment is a basic sensory process and to know of others in the environment is a basic communicative process. As an autonomous agent, if your activity in that environment then effects the state of others in that space then it is a communicative act to sense the output of the other entities so as to be able to self-regulate, whether competitively or collaboratively. This self-regulatory loop is a basic characteristic of a living systems wherein the dynamics of the system have to be brought into consideration by the evolving entity for self-maintenance and the continued functionality of the environment. Selection pressure must depend on the dynamics of organism behavior and Ray and Hart [7] provide an excellent example of the failures that can result if this condition is not recognised in their comments about "mob behaviour" when one of the available niches in the environment is vastly superior to the rest and all organisms attempt to, in the case of networked Tierra, "migrate" to that most attractive context.

In conclusion

Sensing what is in the environment and communication with other like creatures in that environment are both essential aspects of even the most primitive objects of bio-logic. In any simulation of bio-logic and evolution as a process both sensory and communicative processes are a necessary inclusion. Further, intentionality is a necessary aspect of the autonomous behaviour of any entities in biological and in simulated environments. Though much of the work in AL is more about programming technologies than about the exploration of living systems where there is a crossover these three factors should be kept in sight.

References:.

[1] Bateson, G. (1973) *Steps to an Ecology of Mind*. Paladin Books.
[2] Jones, S. (1999) "On Qualia and Information" paper read at the Towards a Science of Consciousness conference held in Tokyo, 1999. (available from sjones@culture.com.au)
[3] Black, I.B. (1991) *Information in the Brain. A Molecular Perspective*. MIT Press, Cambridge, Mass.
[4] Reynolds, C.W. (1987) Flocks, Herds, and Schools: A Distributed Behavioral Model" *Computer Graphics 21, (4) Siggraph '87 Conference Proceedings*, pp25-34.
[5] Werner, G.M. and Dyer, M.G. (1991) "Evolution of Communication in Artificial Orgasnisms" in: Langton, Taylor, Farmer and Rasmussen, eds. *Artificial Life II*. Vol X in Santa Fe Institute Studies in the Sciences of Complexity, Addison Wesley.
[6] Terzopoulos, D., Rabie, T. and Grzeszczuk, R. (1996) "Perception and Learning in Artificial Animals" *Artficial Life V*.
[7] Ray, T.S. (1997) "Evolution as Artist" in: *Art@Science*, Sommerer C., Mignonneau L. (eds), pp81-91. Springer Vienna/New York.
[8] Sommerer, C. and Mignonneau, L. (1997) "Art as a Living System" in: *Art@Science*, Sommerer C., Mignonneau L. (eds), pp81-91. Springer Vienna/New York.
[9] Ray, T.S. and Hart, J. (1998) "Evolution of Differentiated Multi-threaded Digital Organisms" in: *Artificial Life VI Proceedings*, C.Adami, R.K.Belew, H.Kitano, and C.E.Taylor [eds.], pp295-304. The MIT Press, Cambridge.
<http://www.hip.atr.co.jp/~ray/>

Minimum Density of Functional Proteins to Make a System Evolvable

Hideaki Suzuki

ATR Human Information Processing Research Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan
hsuzuki@hip.atr.co.jp

To make an evolutionary system highly evolvable, the genotype space of a protein must be occupied by a number of functional proteins. In addition, these proteins have to be interconnected to each other so that a mutation may be able to create another functional protein and explore the genotype space. The genotype space of the fixed length protein is mathematically analyzed to quantitatively formulate this condition. A graph whose node represents a set of adjacent genotypes mapped to the same phenotype and whose edge represents a mutational transition between a pair of nodes is introduced. We then apply the random graph theory to this graph and formulate the minimum density of functional proteins for high connectivity of the graph. The minimum density is approximately the reciprocal of the product of the average number of adjacent genotypes mapped to the same phenotype and the number of different genotypes created from one genotype through a unit mutational step. The derived formula is tested using data for a fictional two-dimensional protein model.

Keywords: *evolvability, functionality, protein, genotype space, mutation, random graph theory*

1 Introduction

Proteins are fundamental functional units of living organisms. Evolvability (the possibility of various kinds of evolution) of living things is greatly dependent upon the evolvability of proteins; hence, to enhance the evolvability of an evolutionary system, we have to augment the possible evolution of various proteins. The evolvability of protein molecules has been argued by several authors to date. About thirty years ago, Maynard-Smith [1] argued that 'if evolution by natural selection is to occur, functional proteins must form a continuous network which can be traversed by unit mutational steps without passing through nonfunctional intermediates'. In 1991, Lipman et al. [2] conducted numerical experiments using a two-dimensional conformation model of artificial proteins proposed by Lau et al. [3, 4]. Lipman et al. confirmed that the fictional proteins

satisfy Maynard-Smith's condition. As these authors pointed out, the evolvability of proteins is largely determined by the connectivity of functional genotypes in the protein genotype space. Since non-functional proteins that quickly die out cannot be fixed in the population, mutations, that is, slight modifications of genotypes, cannot search for various functional genotypes if functional genotypes are sparsely distributed in the genotype space. For high evolvability, functional proteins have to be densely distributed so that functional genotypes can be interconnected.

Based on this argument, this paper mathematically analyzes the connectivity of functional genotypes and estimates the minimum density of functional proteins that will make a system evolvable. The protein genotype is represented by a sequence of I alphabets chosen among K different alphabets (both are fixed numbers). As mutational modifications, only substitutions of alphabets are considered, and if a protein genotype (a sequence of alphabets) can be changed into another by substitution, these genotypes are called 'adjacent' in the genotype space. The analysis uses a *mutation graph* whose node corresponds to a set of adjacent genotypes mapped to the same phenotype (hereafter such a set is called a *blob*) and whose undirected edge represents a mutational transition between a pair of blobs. The random graph theory [5, 6] states that several characteristics, such as connectivity, the presence of a giant component (connected subgraph), etc., of a random graph dramatically change when the number of edges passes particular threshold values. This result is applied to the mutation graph, and the minimum density of functional proteins is formulated as a threshold density above which a giant component emerges in the mutation graph. The derived formula is numerically tested with Lau et al.'s fictional proteins [3].

The organization of the paper is as follows. In Sect. 2, we formally introduce the mutation graph and derive the paper's main result, the formula for the minimum density of functional proteins. Next, in Sect. 3, the established formula is tested using the model proteins of Lau et al. In Section 4, the advantages and limita-

tions of the formula are described. Section 5 discusses relations to other research on evolvability and future research plans.

2 Mutation graph and its connectivity

Let I alphabets of a sequence be simply classified into two groups: I_s 'substitutable' ones and $I - I_s$ 'unsubstitutable' ones. A blob, a set of adjacent genotypes that have the same phenotype, is a set of sequences comprising I_s volatile alphabets and $I - I_s$ fixed alphabets. If we define K_s as the number of alphabets that can be substituted for a volatile alphabet without changing functionality, the size (number of included genotypes) of a blob is represented by $K_s^{I_s}$. The basic assuptions used in the analysis are as follows.

Assump. 1: The size of a blob is constant. (Both I_s and K_s are fixed numbers for all blobs.)

Assump. 2: Blobs are uniformly distributed in the genotype space.

Assump. 3: A mutation only replaces an alphabet with another alphabet. Modifications causing changes in sequence length (a duplication or a deletion) are not considered.

To analyze the connectivity between blobs, we introduce the mutation graph, which is defined as a graph whose node corresponds a blob and whose undirected edges connect pairs of interconnected blobs (Fig. 1). Let N be the number of blobs (number of nodes of the mutation graph), M be the number of edges of the mutation graph, and ρ be the density of functional genotypes in the protein genotype space (ratio of functional genotypes compared to all genotypes). ρ is expressed as

$$\rho = N \cdot \frac{K_s^{I_s}}{K^I}. \tag{1}$$

Our first goal is to derive a formula for M from the probability of two randomly chosen blobs being interconnected. Hereafter the author refers to a position for an alphabet as a *locus*. (Note that this definition for a locus is different from that in biology.) Since the probability of an arbitrarily chosen alphabet not being included in K_s alphabets is $1 - K_s/K$, when $K_s \ll K$, the probability of two independent sets of K_s alphabets having one or more common alphabets at a locus is approximately given by

A blob (a set of adjacent genotypes with the same phenotype)
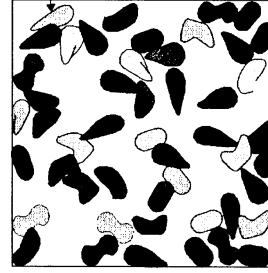
Connected components
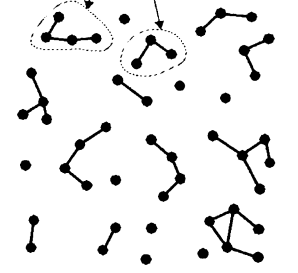
(a) Protein genotype space (b) Mutation graph

Figure 1: (a) Symbolized figures of protein genotype space, and (b) corresponding mutation graph. Various hatching patterns of blobs represent different kinds of phenotypes. Remaining non-hatched (white) regions are non-functional. A node pair of mutation graph is connected by an edge if and only if corresponding blobs are interconnected in genotype space.

$$1 - \left(1 - \frac{K_s}{K}\right)^{K_s} \simeq \frac{K_s^2}{K}. \tag{2}$$

Let us consider two randomly chosen blobs. When comparing alphabets on each blob, I loci in a sequence are classified into three groups: I_{ss} loci in which both blobs have substitutable alphabets, I_{su} loci in which one blob has substitutable alphabets and the other has unsubstitutable alphabets, and I_{uu} loci in which both blobs have unsubstitutable alphabets. The averages of these three numbers are

$$\overline{I_{ss}} = I \times \left(\frac{I_s}{I}\right)^2, \tag{3a}$$

$$\overline{I_{su}} = I \times 2\frac{I_s}{I}\left(1 - \frac{I_s}{I}\right), \tag{3b}$$

$$\overline{I_{uu}} = I \times \left(1 - \frac{I_s}{I}\right)^2. \tag{3c}$$

Here $\overline{I_{ss}} + \overline{I_{su}} + \overline{I_{uu}} = I$. The mutation graph's edge connects a pair of nodes when the corresponding blobs interconnect, or in other words, when the corresponding blobs overlap or are adjacent. Let P_{ovl} be the probability of two randomly chosen blobs overlapping (this actually does not happen, but we can estimate this possibility on account of Assump. 2), and P_{adj} be the probability of two randomly chosen blobs being adjacent. When two blobs overlap, they will have one or

more common alphabets at every locus. Thus, using Eq. (2) and Eqs. (3a)~(3c), P_{ovl} is formulated as

$$
\begin{aligned}
P_{\text{ovl}} &= \left(\frac{K_s^2}{K}\right)^{\overline{I_{\text{ss}}}} \cdot \left(\frac{K_s}{K}\right)^{\overline{I_{\text{su}}}} \cdot \left(\frac{1}{K}\right)^{\overline{I_{\text{uu}}}} \\
&= \frac{K_s^{2\overline{I_{\text{ss}}}+\overline{I_{\text{su}}}}}{K^I} = \frac{K_s^{2I_{\text{s}}}}{K^I}.
\end{aligned} \tag{4}
$$

When two blobs are adjacent, on the other hand, they will have one or more common alphabets at $I-1$ loci and no common alphabet at one remaining locus. Accordingly, using a binomial distribution, P_{adj} is formulated as

$$
\begin{aligned}
P_{\text{adj}} \\
= \overline{I_{\text{ss}}}\left(\frac{K_s^2}{K}\right)^{\overline{I_{\text{ss}}}-1}\left(1-\frac{K_s^2}{K}\right) \cdot \left(\frac{K_s}{K}\right)^{\overline{I_{\text{su}}}} \cdot \left(\frac{1}{K}\right)^{\overline{I_{\text{uu}}}} \\
+\overline{I_{\text{su}}}\left(\frac{K_s^2}{K}\right)^{\overline{I_{\text{ss}}}} \cdot \left(\frac{K_s}{K}\right)^{\overline{I_{\text{su}}}-1}\left(1-\frac{K_s}{K}\right) \cdot \left(\frac{1}{K}\right)^{\overline{I_{\text{uu}}}} \\
+\overline{I_{\text{uu}}}\left(\frac{K_s^2}{K}\right)^{\overline{I_{\text{ss}}}} \cdot \left(\frac{K_s}{K}\right)^{\overline{I_{\text{su}}}} \cdot \left(\frac{1}{K}\right)^{\overline{I_{\text{uu}}}-1}\left(1-\frac{1}{K}\right) \\
= P_{\text{ovl}}\left\{\overline{I_{\text{ss}}}\left(\frac{K}{K_s^2}-1\right)+\overline{I_{\text{su}}}\left(\frac{K}{K_s}-1\right)+\overline{I_{\text{uu}}}(K-1)\right\} \\
\simeq P_{\text{ovl}}I(K-1).
\end{aligned} \tag{5}
$$

Approximation $K_s \approx 1$ is used for the final formula. From Eqs. (4) and (5), the expected number of edges of the mutation graph $\overline{M}$ is given by

$$
\begin{aligned}
\overline{M} &= \binom{N}{2} \cdot (P_{\text{ovl}}+P_{\text{adj}}) \\
&\simeq \frac{N^2}{2} \cdot P_{\text{ovl}}\{1+I(K-1)\} \\
&\simeq \frac{N^2}{2} \cdot \frac{K_s^{2I_s}}{K^I} \cdot I(K-1)
\end{aligned} \tag{6}
$$

According to the random graph theory [5, 6], if the edge number is greater than one half of the node number (namely, $M > N/2$), a giant component emerges in a random graph and the majority of nodes are swallowed in that component. This condition is rewritten with Eqs. (1) and (6) as

$$
\begin{aligned}
\frac{N^2}{2} \cdot \frac{K_s^{2I_s}}{K^I} \cdot I(K-1) &> \frac{N}{2} \\
\rho \cdot K_s^{I_s} \cdot I(K-1) &> 1,
\end{aligned} \tag{7}
$$

or defining ρ_c as the critical density of functional proteins,

$$
\rho > \rho_c \equiv \frac{1}{K_s^{I_s} \cdot I(K-1)}. \tag{8}
$$

ρ_c is inversely proportional to both $K_s^{I_s}$ (the blob size) and $I(K-1)$ (the number of genotypes neighboring one genotype in the search space). When $\rho < \rho_c$, the mutation graph is separated into a number of components and a mutation cannot explore the genotype space, resulting in limited protein evolvability. When $\rho > \rho_c$, on the other hand, the majority of functional proteins interconnect and a mutation can easily explore the genotype space, resulting in high protein evolvability.

3 A Numerical example

In this section, Eq. (8) is applied to Lau et al.'s two-dimensional fictional protein model [3]. A genotype of this model is a two-alphabet fixed-length alphabet sequence, and structure is represented by a configuration on two-dimensional grids. A sequence folded into a stable conformation is considered to be functional. Lipman et al. [2] numerically analyzed the genotype space of this model and studied the distribution of functional sequences. According to Fig. 2 in [2], the blob's average size is approximately 3 for this model although it varies considerably. Thus, ρ_c is estimated as

$$
\rho_c \sim \frac{1}{3 \times 19 \times (2-1)} \sim 0.02.
$$

Figure 4 in [2] shows that about 3% of the genotypes are functional for this model; hence, Eq. (8) is satisfied for this fictional protein model.

4 Conclusion

To measure the evolvability of proteins, we analyzed the distribution of functional proteins in the genotype space. The regions in the genotype space that correspond to one phenotype (function) were represented with a number of separated blobs. By assuming a uniform size and probable occurrence of blobs, we formulated the condition for almost all blobs' being connected by mutational transitions. The formula is satisfied if the product of the density of functional genotypes (ρ), the blob size ($K_s^{I_s}$), and the number of genotypes created by a single mutational step from one genotype ($I(K-1)$) is larger than one. Hence, if we can calculate these quantities of an evolutionary system in some numerical method, we can estimate the system's evolvability.

5 Discussion

Recently, by focusing on the secondary structure of RNA molecules, Reidys et al. [7, 8, 9, 10, 11] studied the genotype space of the RNA base sequence and analyzed the connectivity of *neutral networks*. Their neutral network is a graph whose vertices represent genotypes with the same secondary structure and whose edges represent mutational changes between genotypes. They applied the random graph theory to this network and succeeded in deriving a formula for the minimum ratio of neutral mutants to ensure high connectivity of a neutral network. The basic assumptions for the analyses of Reidys et al. and the analyses in this paper are completely different. As Schuster described in his paper [10], Reidys et al. assumed that 'sequences folding into the same structure are (almost) randomly distributed in sequence space' according to the results of inverse folding [12]. In the present paper, on the other hand, genotypes with the same phenotype are assumed to be unevenly distributed, making separated blobs in the genotype space. According to modern biological studies on the conformations of protein molecules, the amino acid sequences that create the same three-dimensional structure are distributed in an *island-like* way in the sequence space [13]. An island corresponds to a blob. To analyze the protein genotype space, we cannot assume a uniform distribution of genotypes with the same phenotype. Instead, an analysis using a blob is a promising way for estimating the system evolvability.

In the future, the author plans to extend the study in the following directions.

- Analyze the protein genotype space using variable-sized blobs (eliminating Assump. 1) and variable-length protein (eliminating Assump. 3).

- Apply the condition to the evolvability estimation of living proteins and the design optimization of ALife systems.

Because such complex organisms as dinosaurs or mammals have evolved in the terrestrial organic life, proteins are considered to satisfy the condition for high evolvability. In the future, we might be able to confirm that the living proteins satisfy Eqs. (7) or (8) by numerically estimating ρ and K_s^{J}s from genetic information data. In order to make an ALife system highly evolvable, the system design must be optimized so that ρ and K_s^{J}s can be increased and the condition is satisfied. This provides a powerful implementation criteria for an ALife system [14].

References

[1] Maynard Smith, J.: Natural selection and the concept of a protein space. Nature, Lond. **225** (1970) 563–564

[2] Lipman, D.J., Wilbur, W.J.: Modelling neutral and selective evolution of protein folding. Proc. R. Soc. Lond. B **245** (1991) 7–11

[3] Lau, F.L., Dill, K.A.: A lattice statistical mechanics model of the conformational and sequence spaces of proteins. Macromolecules **22** (1989) 3986–3997

[4] Lau, F.L., Dill, K.A.: Theory for protein mutability and biogenesis. Proc. Natn. Acad. Sci. U.S.A **87** (1990) 638–642

[5] Bollobás, B.: Random graphs. Academic Press, London (1985)

[6] Palmer, E.M.: Graphical evolution. John Wiley & Sons, New York (1985)

[7] Reidys, C., Stadler, P.F., Schuster, P.: Genetic properties of combinatory maps - Neutral networks of RNA secondary structures. Bull. Math. Biol. **59** (1997) 339–397 or Santa Fe Working Paper #95-07-058 available at http://www.santafe.edu/sfi/publications/working-papers.html

[8] Reidys, C., Kopp, S., Schuster, P.: Evolutionary optimization of biopolymers and sequence structure maps. In: Langton, C., Shimohara, K. (eds.): Artificial Life V: Proceedings of the Fifth International Conference on Artificial Life. MIT Press, Cambridge, MA (1997) 379–386

[9] Reidys, C.: Random induced subgraphs of generalized n-cubes. Adv. in Appl. Math. **19** (1997) 360–377

[10] Schuster, P.: Landscapes and molecular evolution. Physica D **107** (1997) 351–365 or Santa Fe Working Paper #96-07-047 available at http://www.santafe.edu/sfi/publications/working-papers.html

[11] Reidys, C., Forst, C.V., Schuster, P.: Replication and mutation on neutral networks. Submitted to Bull. of Math. Biol., or Santa Fe Working Paper #98-04-036 available at http://www.santafe.edu/sfi/publications/working-papers.html

[12] Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoeffer, L.S., Tacker, M., Schuster, P.: Fast folding and comparison of RNA secondary structures. Monatshefte für Chemie **125** (1994) 167-188

[13] Nishikawa, K.: Island hypothesis: Protein distribution in the sequence space. Viva Origino **21** (1993) 91-102

[14] Suzuki, H., Ray, T.S.: Conditions to Facilitate the Evolvability of Digital Proteins. To be published in: JCIS 2000 (the Fifth Joint Conference on Information Sciences) Proceedings

Diffusible immortal ALife rarely exterminate diffusible mortal ALife in one finite, heterogeneous ecosystem.

Tadao Maekawa [1], Osamu Ueno [2, 3], Emi Nishina [4, 5], Norie Kawai [3, 5],
Katsunori Shimohara [5], and Tsutomu Oohashi [5]

[1] ATR Media Integration & Communications Research Laboratories, JAPAN
[2] Department of Biochemistry, Gifu University School of Medicine, JAPAN
[3] Foundation for Advancement of International Science, JAPAN
[4] National Institute of Multimedia Education, JAPAN
[5] ATR Human Information Processing Research Laboratories, JAPAN
Contact: maekawa@mic.atr.co.jp

Keywords: Programmed Self-Decomposition, PSD model, death, ecosystem, computer simulation, SIVA

Abstract

We have proposed a theoretical hypothesis of the programmed self-decomposition (PSD) model [1-3] postulating that a self-decomposition mechanism is programmed into all living organisms on earth, and that it contributes to the material as well as the spatial restoration of the ecosystem. In a series of computer simulation studies, we reported that the PSD mechanism contributed to the restoration of a virtual ecosystem and to the evolutionary adaptation of virtual life forms. In addition, the virtual mortal life forms based on the PSD model were not exterminated by the immortal ones in almost all cases even when both the mortal and immortal ones started their lives in the same ecosystem having finite spaces and materials with a heterogeneous distribution. However, there is a possibility that the previous simulation conditions might have been more disadvantageous to the immortal life forms than to the mortal ones because the immortal could not proliferate any more when they eternally occupied their immediate spaces. In the present study, we explicitly addressed this possibility by investigating an additional simulation condition in which both the mortal and immortal life forms were diffusible. The results showed that the mortal life forms were not exterminated in almost all cases.

1. "Programmed Self-Decomposition" Model
1.1 Restoration Mechanism of the Terrestrial Ecosystem

We try to reveal the biological significance of "death." We suppose that death contributes to the restoration of the ecosystem and to evolutionary advancement, and that death itself might be obtained as a fruit of evolution.

The terrestrial ecosystem can be characterized as being almost closed, in that space and substance are both limited. Accordingly, to maintain the stability of that terrestrial life form activity, the space and substance that are removed from the environment by the activity of the life form itself have to be returned to the environment.

The mechanism of restoring the terrestrial ecosystem to its original state has conventionally been explained by the principle of biological circulation called the food chain -- mutually giving oneself as food to other living organisms and depending upon metabolic activity to attain the restoration of the environment as a whole [4]. We have set up a new hypothesis that is complementary to that of the food chain. In the present terrestrial ecosystem, while such a mechanism for restoring the environment to its original state is occurring, another hidden mechanism is fundamentally built into every living organism by which each cell of that organism positively decomposes itself so as to contribute to the restoration of the environment. we call this phenomenon of decomposing oneself as "self-decomposition."

1.2 Theoretical Definition of Self-Decomposition

We have developed a new concept of a self-reproductive, self-decomposable automaton [1] through a modification of von Neumann's self-reproductive automaton model [3]. It can be summarized as follows:
(1) Automaton A constructs another automaton according to instruction I;
(2) Automaton B makes a copy of instruction I;
(3) Mechanism C combines A and B and functions as follows:
   (3-1) let A construct another automaton according to I ;
   (3-2) let B make a copy of I and insert it into the automaton constructed above ;
   (3-3) separate the new automaton from the system A + B + C;
(4) Automaton D consists of A + B + C;
(5) Instruction I_D describes automaton D;
(6) Automaton E consists of $D + I_D$, which can reproduce itself;
(7) Instruction I_{D+F} describes automaton D plus another given automaton F;

(8) Automaton E_F consists of D + $I_{D + F}$, which can reproduce itself and construct another automaton F.

This model continues to self-reproduce as long as there is sufficient space, material and energy, and its structure remains intact unless it is attacked by external forces. It is immortal. On the contrary, living organisms on earth die without exception, and unless special measures are taken, they degrade into components after death. This is the essential difference between von Neumann's model and terrestrial life. So taking into account the ideas presented here, we have developed the following model, using von Neumann's model as a prototype, into which the process of "death and decomposition" is already programmed. This new model can be expressed as a variation of von Neumann's self-reproductive automaton E_F. To be more specific:

(a) Automaton FZ, which has the ability to disjoint the whole system into component elements, is a modular subsystem comparable to von Neumann's automaton F.

(b) Instruction $I_{D + FZ}$ describes automaton D plus automaton FZ.

(c) Automaton E_{FZ} is a system comparable to von Neumann's automaton E_F whose instruction I_{D+F} is replaced by instruction $I_{D + FZ}$.

(d) Automaton G is a system composed of E_{FZ} and FZ, viz., D + FZ + $I_{D + FZ}$.

This system G can reproduce itself, and therefore makes FZ a subsystem within the system. FZ has the ability to disjoint G into finite elements. These elements are sized and structured in such a way that the entire ecosystem that G belongs to may take advantage of them collectively. FZ's mode of action can be one of the following three:

(1) Its production is normally restricted. With the input of a particular message, however, the restriction is lifted and the production activated.

(2) Its operation is normally restricted. With the input of a particular message, however, the restriction is lifted and the operation activated.

(3) (1) and (2) together.

Furthermore, if, after a certain amount of time has passed or a certain set of events has occurred, there is still no message input to trigger an action, (1), (2), or (3) will happen automatically. The "certain message" referred to above is provided, evidently, when it becomes impossible for G to further multiply itself or to maintain its structure, that is, when it is on the verge of extinction. If G is not given such a message for a certain period of time, G automatically puts FZ into operation. Therefore, we regard G as an automaton having a program of spontaneous and

inevitable "death and decomposition" installed *a priori*.

System G composed as described above not only reproduces itself, but also has the ability to put an end to its own life and to return to its origin, that is, to contribute to restoring the ecosystem to its original state. We tentatively call this system a "self-reproductio~self-decomposition (SRSD) system" and call the theoretical model under discussion a "programmed self-decomposition (PSD) model."

The above concept was first published in Japanese in 1987 [1]. Independently, Todd presented a concept of artificial death in 1993 [6]. He did not address the spontaneity and purposefulness of the "death" and, therefore, appeared to investigate an essentially different concept, such as autolysis, from our PSD model. Nevertheless, it is noteworthy that he pointed to the significance of death for living organisms.

2 A Study on Virtual Life -- "SIVA-3"
2.1 Motivation

We adopted two approaches in order to test this model: biological experiments using actual life forms and computer simulations using virtual life forms.

We developed a simulator series called "SIVA (Simulator for Individuals of Virtual Automata)" in order to investigate the effectiveness of "death" [2,3]. Recently, another investigation with CA is being conducted by Sayama [7] based on a similar concept. SIVA is distinguished from the other ALife simulators by its finite and heterogeneous environmental conditions and by the rational life activities based on the above automaton model, which are designed to resemble the terrestrial ecosystem and life forms, respectively. Thus, it seems reasonable to say that experimental simulations using SIVA have provided a detailed understanding of the life system based on more realistic simulations than others. We found that a mortal life based on the PSD model has a fundamental advantage over an immortal life when they live independently in two similarly conditioned environments. Since an immortal life continues to occupy space and consume substances eternally, another life, including one of its own species, may be difficult to proliferate in the same ecosystem. On the other hand, the mortal life can contribute to restoring the simulated ecosystem and continuously reuse its limited space and substances. It also provides a greater opportunity for evolutionary adaptation [2]. In addition, we showed that, contrary to expectations, the mortal life forms were not exterminated by the immortal ones, but gained advantages over the immortal ones in many cases, when both the

mortal and immortal life forms started their lives in the same ecosystem [3]. These simulations suggested the biological significance of the PSD mechanism and its tenacity and advantage in natural selection. However, each individual of both life forms was not movable and was fixed to the point where it was born in the simulations. It is difficult to deny the possibility that this experimental condition might be unfairly disadvantageous to the immortal life forms, because the spaces adjacent to the immortal life form were eternally occupied by other individuals and thus they might fail to proliferate due to the lack of space. In the present study, we allowed all individuals, both mortal and immortal life forms, to diffuse randomly and to more easily find spaces for reproduction. Under this simulation condition, we examined whether the extermination of the mortal life forms by the immortal ones could occur.

2.2 Design of the Simulations Environment

SIVA-3 was designed as previously described [2]. The virtual space in SIVA-3 is assumed to be a 2-dimensional lattice of 128x128 pixels. One pixel is defined as a spatial unit that one individual of virtual life must occupy to exist. The virtual substances, of which the ecosystem and the artificial lives in SIVA-3 are composed, are restricted to four types of elements. Environmental finiteness and heterogeneity are introduced into 256 (16 x 16) spatial blocks (each consisting of 8 x 8 pixels), on the basis of the quantities of substances (Figure 1). That is, the quantities of the substance in the spatial blocks differ. These conditions range from those with maximum conformity for primitive life, to those with minimum conformity, that is, a condition in which it is impossible for primitive life to survive or reproduce itself unless it adapts to the condition of taking many evolutionary steps. The substances were allowed to diffuse among the spatial blocks so as to maintain this type of heterogeneity. SIVA-3 can also make heterogeneity based on temperature, but this was not adopted in the current simulations in order to make the simulative conditions simpler. Sufficient energy for life activities is assumed to be given [2, 3].

Figure 1 Initial distribution of the elemental substances A, B, C and D in SIVA-3

Life

A mortal PSD life form based on the SRSD system

was designed, that is, a modular subsystem, which has the ability to disjoint the whole system into component elements, added on the von Neumann's self-reproductive automaton. An immortal non-PSD life was designed by destroying the self-decomposing ability of the mortal life form. They consist of VC (Instruction I including I_D and $I_{D + FZ}$) and VP (Automata including A, B, FZ and Mechanism C). Each of the VCs and VPs was made from the elemental substances A, B, C and D, and was designed as having plural choices of formation. Changes in formation caused by mutation may make a newborn individual more conformable to another new spatial block in which other quantities of substances were buried [2, 3].

In our previous simulations, since an individual of both life forms was immovable and eternally stayed at the place where it was born, the older individuals of the immortal life form monotonously occupied the spaces most suitable to a primitive life form. The reduction of adjacent spaces for reproduction caused serious damage to other individuals, both mortal and immortal. This situation was a natural consequence caused by the immortal life form. However, there is a possibility that this situation might be unfairly more disadvantageous to the immortal life forms than to the mortal life forms. In consideration of this point, we designed an additional condition that allowed each individual to diffuse to the neighboring pixel in a random direction every 1 time count (TC) of a simulation. Any individual could reach anywhere in the SIVA world if it could spend enough time. An individual was allowed to be active within the areas of an 8-pixel-radius distance from its birth position, and outside this area all of its activities, including self-decomposition, were suspended. Such a condition would be expected to avoid extreme concentration of the population and reduce the limitation on reproduction due to the monotonous occupancy of adjacent spaces by the older immortal individuals. This effect appeared to be more remarkable for the immortal life forms than for the mortal ones. Moreover, this condition resembles actual monocellular organisms in water that are not immovable, and enables a more realistic simulation with respect to the actual terrestrial ecosystem.

An internal status variable UNCONF (non-negative integer) was installed as a measure of conformity with the environment [3]. The default value of UNCONF is 0 when there is complete conformity with the environment. When a command to reproduce a life form fails to be executed (caused by a lack of substances, for example), the individual perceives unconformity with the environment and increases the value of its internal status variable, UNCONF, by 1 per failure. When the value of UNCONF

goes beyond 2, the mortal life form based on the PSD model perceives unacceptable unconformity that exceeds the threshold and activates the decomposing module FZ in order to decompose itself. The mortal life form also starts self-decomposition when its age exceeds 20 TC of a simulation, even if it does not perceive any signal of unconformity.

There are two approaches for adapting to an environment with which the individual does not have conformity. They are evolutionary adaptation and temporary adaptation. The following two global variables were introduced in order to investigate the requirements for the extermination of mortal life forms by immortal life forms [3].

Mutation Rate. At a probability of *mis-VC-rate* (between 0.0 - 1.0) per character of VC, substitution will occur when the VC is copied. Masking against mutation was applied to the VCs of those individuals corresponding to automaton A, B, C, or FZ, so as to prevent the individuals from losing the function of that portion.
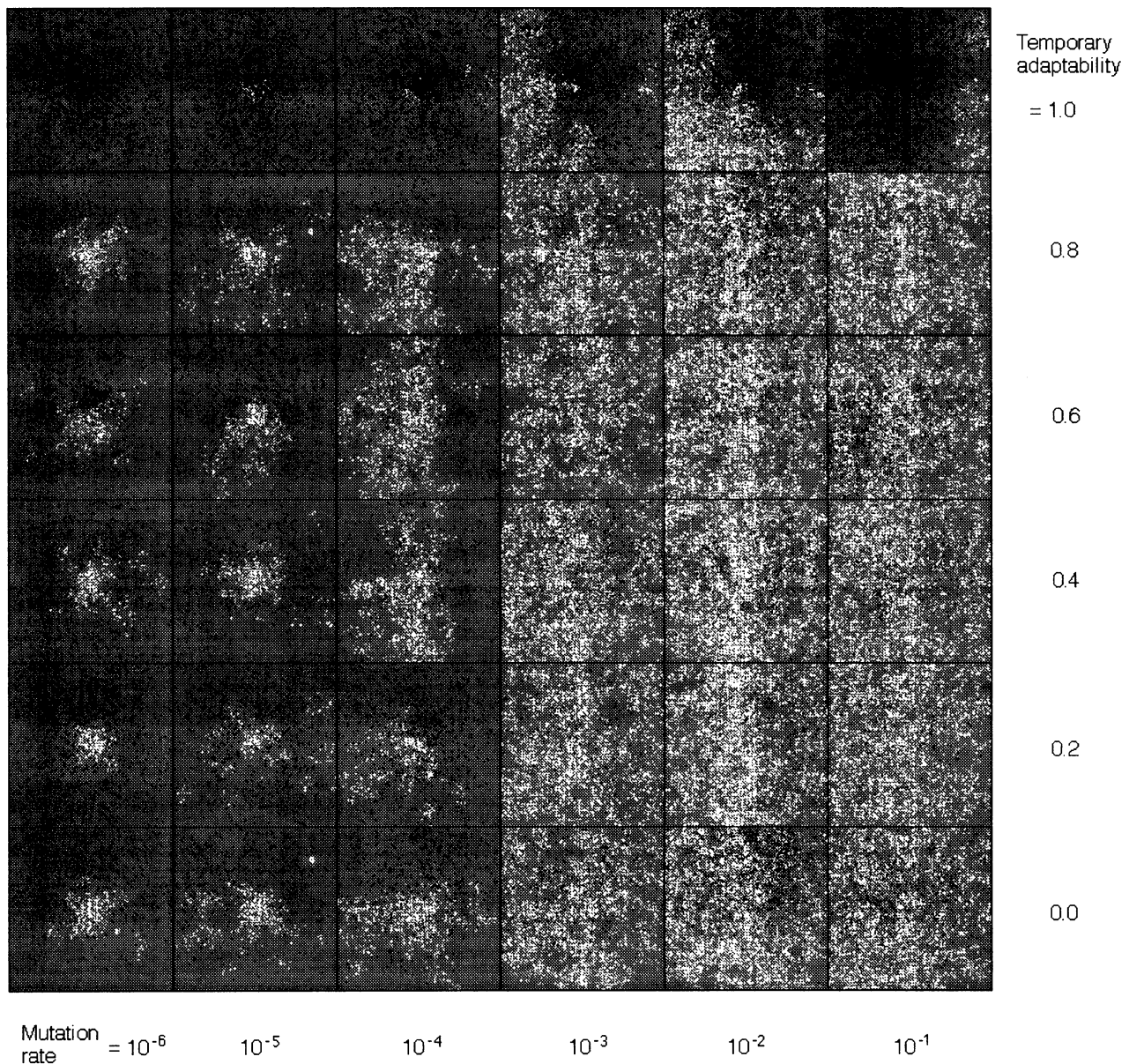
Figure 2   Typical distributions of virtual life forms at 1000 TC. The figures are organized according to mutation rate (*mis-VC-rate*) in the vertical direction and according to the temporary adaptability (TEMP-ADAPT) in the horizontal direction. The black dots denote the immortal individuals and the white dots the mortal ones.

Temporary adaptability. TEMP-ADAPT (between 0.0 - 1.0) was designed as the degree of temporary adaptability for rearranging current activities and getting through a situation without evolutionary adaptation. The probability that each of the commands for the self-reproductive activities will execute was calculated as TEMP-ADAPT raised to the power of UNCONF in this investigation. In the case of low temporary adaptability, the higher the accumulated failure (the larger the value of UNCONF) the less executable are the commands.

2.3 Simulations

Methods

Both an individual immortal life form and a mortal life form based on the PSD model were seeded in the center of the same ecosystem in SIVA-3. There were 36 simulations conducted in all, including 6 cases of temporary adaptability (TEMP-ADAPT: 0.0, 0.2, 0.4, 0.6, 0.8, and 1.0) for 6 cases of mutation rate (*mis-VC-rate*: 1x10-6, 1x10-5, 1x10-4, 1x10-3, 1x10-2, and 1x10-1). Each simulation was carried out 10 times from TC (Time Count) = 0 to TC = 1000. The extermination ratio of mortal life forms was calculated for each of the 36 cases. The population of individuals and accumulated frequency of reproduction, decomposition and mutation were counted for every TC in each of the 36 cases.

Results

The diffusible condition introduced a wider distribution of individuals over almost all of the blocks of the virtual environment, as expected. The higher mutation rate introduced a wider distribution of the mortal life forms

(Figure 2). When additional simulations were conducted with a longer duration and lower mutation rates, it was observed that the distributions of the mortal life forms were similar to those at 1000 TC with the higher mutation rates.

In the 36 cases that were tested, the inevitable extermination of the mortal life forms never occurred. Only the 3 cases with the highest temporary adaptability (TEMP-ADAPT: 1.0) exceeded an extermination ratio of 1/2. In the other 33 cases, the mortal life forms were never or only rarely exterminated (0 for 28 cases, and 0.1 in 5 cases)(Figure 3).

The distributions of the individuals of the two life forms were frequently intermixed. However, there was a tendency for the immortal individuals to gather and form colonies in the cases of extremely high temporary adaptability compared to the other cases. Almost all of the extermination of mortal life forms occurred when the mortal individuals could not escape from the colonies of the immortal individuals (Figure 2).

Typical time courses for the number of individuals, frequency of reproduction and decomposition, and accumulated frequency of mutation are shown for each type of life form in Figure 4. The activities of the mortal individuals were overwhelmingly higher than those of the immortal individuals (the number of individuals: 7.7 times; frequency of mutation: 155 times, at 1000 TC).

Figure 3 Extermination ratios of the mortal life forms by the immortal life forms in all 36 cases. The number of simulations in which all individual mortal life forms were exterminated is shown. The case in which the number of mortal individuals was less than 20 but more than zero was counted as 0.5.
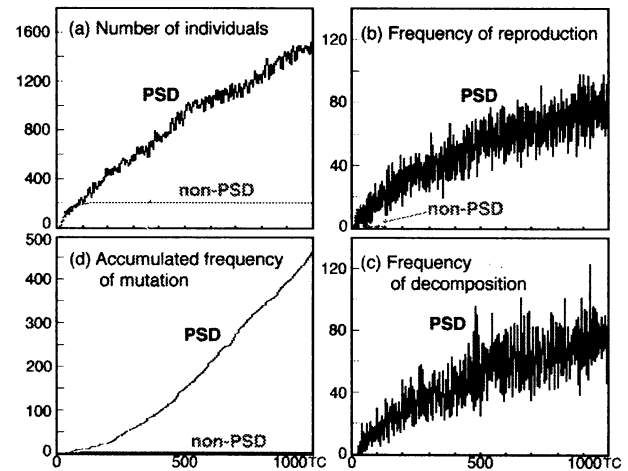
Figure 4 Typical time courses of various activities by the mortal and immortal life forms from 0 TC to 1000 TC (*mis-VC-rate*: 10-4; TEMP-ADAPT: 0.4). (a) The number of individuals, (b) frequencies of reproduction, (c) frequency of decomposition (only for the mortal individuals), and (d) accumulated frequency of mutation.

3. Discussion

The present simulations showed that the mortal life forms were not exterminated in almost all cases, but gained advantages over the immortal life forms in many cases, even when all individuals were diffusible in a finite and heterogeneous virtual environment. Extermination occurred only in special conditions (i.e., the highest temporary adaptability (TEMP-ADAPT: 1.0)). This tendency compares well with the results of the previous simulation in which individual life forms were immovable, although the extermination ratios of the mortal life forms in the present, diffusible condition were even lower than those in the previous studies with an immovable condition [3]. This finding suggests, contrary to expectations, that the diffusible condition had a more advantageous effect on the mortal life forms than on the immortal life forms. We suppose that the diffusible condition helped the mortal individuals to escape from areas occupied by the immortal individuals.

As shown in the previous reports [2, 3], the PSD mechanism contributed to the restoration of the virtual ecosystem and to the evolutionary adaptation in the present simulations. Consequently, the species of the mortal life forms became more abundant in a broader area of the heterogeneous virtual ecosystem than the immortal life forms. The complexity of the ecosystem was also realized.

It is noteworthy that these findings were obtained in simulations in which the mortal and immortal life forms co-existed in the same finite, heterogeneous ecosystem. An argument against our concept of the genetically programmed death stated that such an inherited quality would be exterminated by natural selection and that, therefore, such a concept was nonsense. However, the series of simulations revealed the possibility that PSD-based mortal life forms could seize an advantage over immortal life forms in a finite, heterogeneous ecosystem under certain conditions, which appeared rather general and realistic. We envisage the possibility that PSD-based mortal life forms could survive when they evolve from the more primitive, immortal life forms. On the other hand,

we can imagine that PSD-based mortal life forms would not be exterminated even when immortal life forms evolve by losing the PSD mechanism by mutation.

4. Conclusion

The present simulations in a finite and heterogeneous environment revealed that PSD-based mortal life forms were rarely exterminated by immortal life forms, even when all individuals were diffusible and could find space for reproduction more easily.

References

[1] Oohashi T., Nakata D., Kikuta T. and Murakami K. (1987) Programmed self-decomposition model, Kagakukisoron kenkyuu, Vol. 18, No. 2, pp. 79-87 (in Japanese)

[2] Oohashi T., Sayama H., Ueno O. and Maekawa T. (1996) Artificial life based on programmed self-decomposition model, ATR Technical Report TR-H-198

[3] Oohashi T., Maekawa T., Ueno O., Nishina E. and Kawai N. (1999) Requirements for immortal ALife to exterminate mortal ALife in one finite, heterogeneous ecosystem, 5th European Conference on Artificial Life (ECAL99), pp. 49-53.

[4] Odum E. P. (1971) Fundamentals of Ecology, Third ed., W. B. Saunders Company, Philadelphia, pp. 104-105

[5] Von Neumann J. (1948) The general and logical theory of automata. In cerebral mechanisms in behavior - The Hixon symposium (Jeffress L. A. (ed.), 1951) John Wiley & Sons, Inc., New York, pp. 1-41

[6] Todd, P. M. (1993) Artificial Death, Second European Conference on Artificial Life (ECAL93), pp. 1048-1059.

[7] Sayama, H. (1998) Introduction of Structural Dissolution into Langton's Self-Reproducing Loop, Artificial Life VI, MIT Press, pp. 114-122.

Evolutionary advantage of Self-Decomposition Mechanism

Tadao Maekawa [1], Osamu Ueno [2, 3], Emi Nishina [4, 5], Norie Kawai [3, 5],
Katsunori Shimohara [5], and Tsutomu Oohashi [5]

[1] ATR Media Integration & Communications Research Laboratories, JAPAN
[2] Department of Biochemistry, Gifu University School of Medicine, JAPAN
[3] Foundation for Advancement of International Science, JAPAN
[4] National Institute of Multimedia Education, JAPAN
[5] ATR Human Information Processing Research Laboratories, JAPAN
Contact: maekawa@mic.atr.co.jp

Keywords: Programmed Self-Decomposition, PSD model, death, ecosystem, computer simulation, SIVA

Abstract

We have proposed a theoretical hypothesis of the programmed self-decomposition (PSD) model [1-4] postulating that a self-decomposition mechanism is programmed into all living organisms on earth, and that it contributes to the material as well as the spatial restoration of the ecosystem. In a series of computer simulation studies, we reported that the PSD mechanism contributed to the restoration of a virtual ecosystem and to the evolutionary adaptation of virtual life forms. In addition, the virtual mortal life forms based on the PSD model were not exterminated by the immortal ones in almost all cases even when both the mortal and immortal ones started their lives in the same ecosystem having finite spaces and materials with a heterogeneous distribution. In the present study, the possibility of the evolutionary appearance of the PSD mechanism was investigated. The results suggested certain evolutionary advantages of the PSD mechanism.

1. "Programmed Self-Decomposition" Model
1.1 Restoration Mechanism of the Terrestrial Ecosystem

We try to reveal the biological significance of "death." We suppose that death contributes to the restoration of the ecosystem and to evolutionary advancement, and that death itself might be obtained as a fruit of evolution.

The terrestrial ecosystem can be characterized as being almost closed, in that space and substances are both limited. Accordingly, to maintain the stability of that terrestrial life form activity, the space and substances that are removed from the environment by the activity of the life form itself have to be returned to the environment.

The mechanism of restoring the terrestrial ecosystem to its original state has conventionally been explained by the principle of biological circulation called the food chain -- mutually giving oneself as food to other living organisms and depending upon metabolic activity to attain the restoration of the environment as a whole [5]. We have set up a new hypothesis that is complementary to that of the food chain. In the present terrestrial ecosystem, while such a mechanism for restoring the environment to its original state is occurring, another hidden mechanism is fundamentally built into every living organism by which each cell of that organism positively decomposes itself so as to contribute to the restoration of the environment. We call this phenomenon of decomposing oneself "self-decomposition."

1.2 Theoretical Definition of Self-Decomposition

We have developed a concept of a self-reproductive, self-decomposable automaton [1] through a modification of von Neumann's self-reproductive automaton model [6] as follows:

(1) Automaton A constructs another automaton according to instruction I;

(2) Automaton B makes a copy of instruction I;

(3) Mechanism C combines A and B and functions as follows:

 (3-1) let A construct another automaton according to I ;

 (3-2) let B make a copy of I and insert it into the automaton constructed above ;

 (3-3) separate the new automaton from the system A + B + C;

(4) Automaton D consists of A + B + C;

(5) Instruction I_D describes automaton D;

(6) Automaton E consists of D + I_D, which can reproduce itself;

(7) Automaton FZ, which has the ability to disjoint the whole system into component elements, is a modular subsystem comparable to von Neumann's automaton F;

(8) Instruction $I_{D + FZ}$ describes automaton D plus automaton FZ;

(9) Automaton E_F consists of D + $I_{D + FZ}$, which can reproduce itself and construct another automaton FZ;

(10) Automaton G is a system composed of E_{FZ} and FZ, viz., $D + FZ + I_{D+FZ}$.

This system G can reproduce itself, and therefore makes FZ a subsystem within the system. FZ has the ability to disjoint G into finite elements. These elements are sized and structured in such a way that the entire ecosystem that G belongs to may take advantage of them collectively. FZ's mode of action can be one of the following three:

(1) Its production is normally restricted. With the input of a particular message, however, the restriction is lifted and the production activated.

(2) Its operation is normally restricted. With the input of a particular message, however, the restriction is lifted and the operation activated.

(3) (1) and (2) together.

Furthermore, if, after a certain amount of time has passed or a certain set of events has occurred, there is still no message input to trigger an action, (1), (2), or (3) will happen automatically. The "certain message" referred to above is provided, evidently, when it becomes impossible for G to further multiply itself or to maintain its structure, that is, when it is on the verge of extinction. If G is not given such a message for a certain period of time, G automatically puts FZ into operation.

System G composed as described above not only reproduces itself, but also has the ability to put an end to its own life and to return to its origin, that is, to contribute to restoring the ecosystem to its original state. We tentatively call this system a "self-reproductio~self-decomposition (SRSD) system" and call the theoretical model under discussion a "programmed self-decomposition (PSD) model."

2 A Study on Virtual Life -- "SIVA-3"
2.1 Motivation

We adopted two approaches in order to test this model: biological experiments using actual life forms and computer simulations using virtual life forms.

We developed a simulator series called "SIVA (Simulator for Individuals of Virtual Automata)" in order to investigate the effectiveness of "death" [2,3]. SIVA is distinguished from the other ALife simulators by its finite and heterogeneous environmental conditions and by the rational life activities based on the above automaton model, which are designed to resemble the terrestrial ecosystem and life forms, respectively. Recently, some approaches were conducted to investigate the effectiveness of death [7, 8]. It seems reasonable to say that experimental simulations using SIVA have provided a detailed understanding of the life system based on more realistic simulations than others. We found that a mortal life based on the PSD model has a fundamental advantage over an immortal life when they live independently in two similarly conditioned environments. Since an immortal life continues to occupy space and consume substances eternally, another life, including one of its own species, may be difficult to proliferate in the same ecosystem. On the other hand, the mortal life can contribute to restoring the simulated ecosystem and continuously reuse its limited space and substances. It also provides a greater opportunity for evolutionary adaptation [2]. In addition, we showed that, contrary to expectations, the mortal life forms were not exterminated by the immortal ones, but gained advantages over the immortal ones in many cases, when both the mortal and immortal life forms started their lives in the same ecosystem [3, 4]. The possibility should be examined that PSD-based mortal life forms could survive when they evolve from the more primitive, immortal life forms, and that PSD-based mortal life forms would not be exterminated when immortal life forms evolve by losing the PSD mechanism by mutation. We conducted simulations in which the PSD mechanism itself was targeted by evolution.

2.2 Design of the Simulations Environment

SIVA-3 was designed as previously described [2]. The virtual space in SIVA-3 is assumed to be a 2-dimensional lattice of 128x128 pixels. One pixel is defined as a spatial unit that one individual of virtual life must occupy to exist. The virtual substances, of which the ecosystem and the artificial lives in SIVA-3 are composed, are restricted to four types of elements. Environmental finiteness and heterogeneity are introduced into 256 (16 x 16) spatial blocks (each consisting of 8 x 8 pixels), on the basis of the quantities of substances. That is, the quantities of the substance in the spatial blocks differ. These conditions range from those with maximum conformity for primitive life, to those with minimum conformity, that is, a condition in which it is impossible for primitive life to survive or reproduce itself unless it adapts to the condition of taking many evolutionary steps. The substances were allowed to diffuse among the spatial blocks so as to maintain this type of heterogeneity. SIVA-3 can also make heterogeneity based on temperature, but this was not adopted in the current simulations in order to make the simulative conditions simpler. Sufficient energy for life activities is assumed to be given [2, 3].
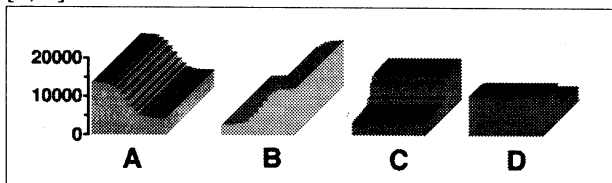
Life

A mortal PSD life form based on the SRSD system

was designed, that is, a modular subsystem, which has the ability to disjoint the whole system into component elements, was added onto von Neumann's self-reproductive automaton. One individual consists of VC (Instruction I including I_D and I_{D+FZ}) and VP (Automata including A, B, FZ and Mechanism C). Each of the VCs and VPs was made from the elemental substances A, B, C and D, and was designed as having plural choices of formation. Changes in formation caused by mutation may make a newborn individual more conformable to another new spatial block in which other quantities of substances were buried [2, 3]. Each individual diffused to the neighboring pixel in a random direction every 1 time count (TC) of a simulation. This condition prevented extreme concentration of the population. Any individual could reach anywhere in the SIVA world if it could spend enough time. An individual was allowed to be active within the areas of an 8-pixel-radius distance from its birth position, and outside this area all of its activities, including self-decomposition, were suspended [4].

An internal status variable UNCONF (non-negative integer) was installed as a measure of conformity with the environment. The default value of UNCONF is 0 when there is complete conformity with the environment. When a command to reproduce a life form fails to be executed (caused by a lack of substances, for example), the individual perceives unconformity with the environment and increases the value of its internal status variable, UNCONF, by 1 per failure. When the value of UNCONF goes beyond 2, the mortal life form based on the PSD model perceives unacceptable unconformity that exceeds the threshold and activates the decomposing module FZ in order to decompose itself. The mortal life form also starts self-decomposition when its age exceeds 20 TC of a simulation, even if it does not perceive any signal of unconformity [3].

There are two approaches for adapting to an environment with which the individual does not have conformity. They are evolutionary adaptation and temporary adaptation. The following two global variables were introduced in order to investigate the evolutionary merit or demerit of the PSD mechanism [3].

Mutation Rate. At a probability of *mis-VC-rate* (between 0.0 - 1.0) per character of VC, substitution will occur when the VC is copied. Masking against mutation was applied to the VCs of those individuals corresponding to automaton A, B, C, or FZ, so as to prevent the individuals from losing the function of that portion.

Temporary adaptability. TEMP-ADAPT (between 0.0 - 1.0) was designed as the degree of temporary adaptability for rearranging current activities and getting through a

situation without evolutionary adaptation. The probability that each of the commands for the self-reproductive activities will execute was calculated as TEMP-ADAPT raised to the power of UNCONF in this investigation. In the case of low temporary adaptability, the higher the accumulated failure (the larger the value of UNCONF) the less executable are the commands.

An immortal pre-PSD life was designed by inhibiting the self-decomposing ability of the PSD model based mortal life form which was designed as above. The PSD mechanism presupposes steps of judgment which evaluate factors such as unconformity with the environment and decide whether to start the self-decomposition function or not. One character of VC representing each of the two judging steps was replaced with another character to fix the judgment as "Do not start the self-decomposition function." This pre-PSD individual was considered to possess a kind of precursor of the PSD mechanism. In the case that at least one of these two replaced characters was mutated and that the mutation enabled normal judgment, the offsprings of this individual became able to self-decompose according to the conditions. In the current simulations, the possibility of such a mutation occurring to enable self-decomposition was nearly 1/220 in the occurrence of all the mutations. In contrast, mutations which disabled the PSD mechanism and made a pre-PSD individual from a PSD individual were possible in a lower possibility.

2.3 Simulations

Methods

A pre-PSD individual was seeded in the center of the ecosystem in SIVA-3. There were 36 simulations conducted in all, including 6 cases of temporary adaptability (TEMP-ADAPT: 0.0, 0.2, 0.4, 0.6, 0.8, and 1.0) for 6 cases of mutation rate (*mis-VC-rate*: 1×10^{-6}, 1×10^{-5}, 1×10^{-4}, 1×10^{-3}, 1×10^{-2}, and 1×10^{-1}). Each simulation was carried out 10 times from TC (Time Count) = 0 to TC = 1000. The number of simulation, in which PSD individuals evolved, was counted for each of the 36 cases. The population of individuals and accumulated frequency of reproduction, decomposition and mutation were counted for every TC in each of the 36 cases.

Results

Table 1 shows the number of simulations where the PSD mechanism was obtained by mutation and at least one PSD individual occurred for each of the 36 cases. The higher the mutation rate, in which the larger the number of simulations PSD individuals evolved. The evolution of PSD individuals was observed in almost all simulations

in the case of more than 10^{-2} of *mis-VC-rate* (mutation rate). This evolution did not occur in the case of 10^{-5} and 10^{-6} of *mis-VC-rate*. This was assumed to be because, stochastically, The duration of the simulations was too short and the space was too narrow to realize sufficient reproduction frequency and mutation frequency in the case of the lower mutation rate. PSD life forms were expected to make an evolutionary adaptation even in the lower mutation rate, but the mutation frequency of life forms that did not possess the PSD mechanism was much lower. than that of PSD life forms in the same duration of simulations [4]. Evolution from the pre-PSD life forms, which did not possess the PSD mechanism, was presupposed in the current simulations. Therefore, in order to conduct simulations within a practical time, the longer duration of simulations was cut, and the results in the case of the higher mutation rate, where the evolution was stably observed, were subjected to the following analyses.

Table 1   The number of simulations in which a PSD individual evolved

TA MR	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}	10^{-1}
1.0	0	0	0	4	10	10
0.8	0	0	0	0	10	10
0.6	0	0	0	1	9	10
0.4	0	0	1	1	10	10
0.2	0	0	1	0	10	10
0.0	0	0	1	0	10	10

The number of simulations in which PSD individuals were not exterminated and still survived at TC 1000 were counted and each survival ratio was calculated (Table 2). Cases in which the number of PSD individuals was less than 20 but more than zero were counted as 0.5, the same as in previous investigations [3, 4]. The ratio of PSD individuals that were not exterminated and succeeded in leaving offsprings exceeded 1/2 in almost all cases where PSD individuals evolved.

Table 2   Survival ratio of PSD individuals (at TC 1000)

TA MR	10^{-2}	10^{-1}
1.0	0.7	0.9
0.8	0.6	1.0
0.6	0.28	1.0
0.4	0.3	1.0
0.2	0.6	1.0
0.0	0.65	1.0

The life activities of the surviving PSD individuals were compared with those of pre-PSD individuals in the same conditions. Figure 2 shows the time course of their reproduction frequency and the PSD individuals' decomposition frequency for each TC during the whole simulation. Figure 2 indicates clearly that the life

activities of PSD individuals were overwhelmingly higher than those of pre-PSD individuals. It was also observed that the reproduction and decomposition of PSD individuals had tension, and this suggests a stable and dynamic equilibrium of the state of the group of the individuals.
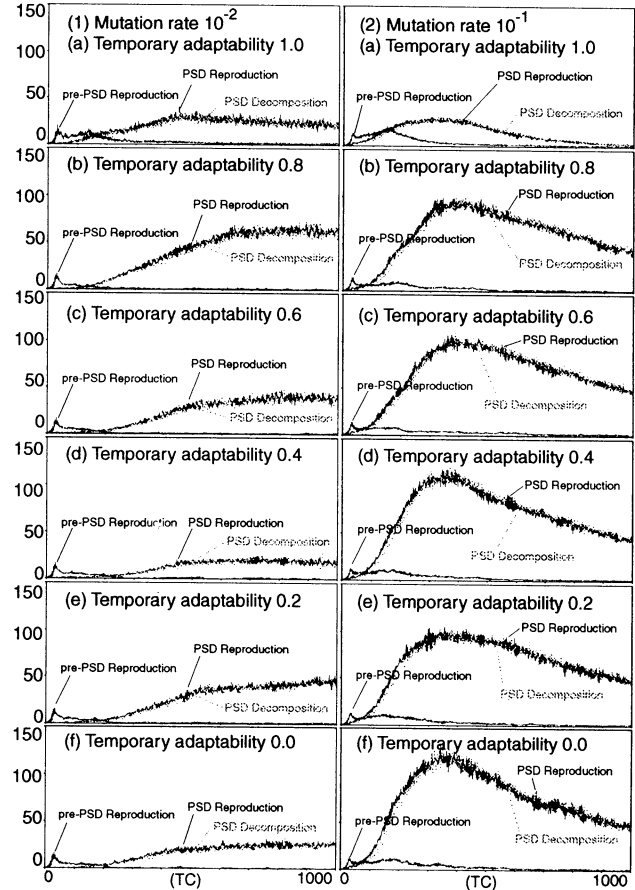
Figure 2   Time course of the average frequency of reproduction and decomposition as representative activities of PSD and pre-PSD individuals. The frequency at each TC was averaged among the simulations for each condition. (1) *mis-VC-rate* 10^{-2} (a) TEMP-ADPT 1.0 (b) TEMP-ADPT 0.8 (c) TEMP-ADPT 0.6 (d) TEMP-ADPT 0.4 (e) TEMP-ADPT 0.2 (f) TEMP-ADPT 0.0 (2) *mis-VC-rate* 10^{-1} (a) TEMP-ADPT 1.0 (b) TEMP-ADPT 0.8 (c) TEMP-ADPT 0.6 (d) TEMP-ADPT 0.4 (e) TEMP-ADPT 0.2 (f) TEMP-ADPT 0.0. Note that the decomposition frequency is observed only in PSD individuals.

3.   Discussion

The present study showed that the PSD-based mortal life forms evolved from the immortal life forms were able to survive in most of the conditions of the simulation. A survival ratio of more than 1/2 and an overwhelmingly high level of activities by PSD individuals were observed under many conditions. These findings suggest that a similar evolutionary phenomenon might occur during the long history of actual living organisms. That is, although we have no evidence to clarify whether the PSD mechanism was installed into terrestrial living organisms

from the beginning or not, the present simulation showed the possibility that the mechanism for death could be obtained by evolution even if such a mechanism was not initially installed in terrestrial life forms.

Immortal virtual life forms also appeared by losing the PSD mechanism due to mutation (PSD-lost immortal life forms). They were observed more frequently in the case of the higher mutation rates. PSD individuals were rarely exterminated even among the pre-PSD and PSD-lost immortal individuals. However, in the case of the highest mutation rate of mis-VC-$rate$ 10^{-1}, the time course of the reproduction and decomposition frequency of PSD individuals showed a gradual decrease toward the TC of 1000 following a peak. It is likely that the increased number of PSD-lost individuals might threaten the PSD individuals more intensively. Although a higher mutation rate could introduce various characteristics and construct an abundant and complex ecosystem, we know that it might also risk introducing adverse characteristics to the actual living organisms. Thus, the ecosystem might become unstable. On this point, it is noteworthy that stably developing ecosystems were evident in the conditions with relatively lower mutation rate (mis-VC-$rate$ 10^{-2} (this seems not to be fully low)), in other words, more possible realistic conditions.

Immortality seems, at first glance, to be a very robust, stout, strong, and ultimate characteristic of any life form. However, immortal life forms have serious weak points, as indicated by Todd [7]. On the other hand, life forms possessing the PSD mechanism can contribute to the restoration of the finite environment where their forefathers had lived. This characteristic can introduce another advantage of the alternation of generations and evolution [see also 2-4]. Moreover, the PSD life forms could construct a stable and dynamic equilibrium state for the group, and easily advance toward unknown frontiers with such a group as a background. PSD individuals differentiated into various species, extended over the virtual ecosystem, and fixed themselves around there. This phenomenon seems to correspond to the "habitat segregation" addressed by Dr. Imanishi [9]. Evolution advanced in the direction of increasing the density of the habitat segregation. Consequently, these results suggest that the PSD mechanisms may be a significant and advantageous strategy for the survival of virtual life forms in a finite and heterogeneous environment. The interaction among individuals and the relationship of the PSD mechanism and food chain should also be addressed in the future.

4. Conclusion

Simulations of ALife with the PSD mechanism in a finite and heterogeneous environment suggest that PSD-based mortal life forms were not exterminated and did survive, when the PSD mechanism evolved from the immortal, primitive life forms, or when immortal life forms appeared evolutionarily by losing the PSD mechanism due to mutation.

References

[1] Oohashi T., Nakata D., Kikuta T. and Murakami K. (1987) Programmed self-decomposition model, Kagakukisoron kenkyuu, Vol. 18, No. 2, pp. 79-87 (in Japanese)

[2] Oohashi, T., Sayama, H., Ueno, O. and Maekawa, T. (1996) Artificial life based on programmed self-decomposition model, ATR Technical Report TR-H-198

[3] Oohashi, T., Maekawa, T., Ueno, O., Nishina, E. and Kawai, N. (1999) Requirements for immortal ALife to exterminate mortal ALife in one finite, heterogeneous ecosystem, 5th European Conference on Artificial Life (ECAL99), pp. 49-53.

[4] Maekawa, T., Ueno, O., Nishina, E., Kawai, N., Shimohara, K., and Oohashi, T. (2000) Diffusible immortal ALife rarely exterminate diffusible mortal ALife in one finite, heterogeneous ecosystem, the 5th international symposium on Artificial Life and Robotics (AROB2000), in this proceedings.

[5] Odum, E. P. (1971) Fundamentals of Ecology, Third ed., W. B. Saunders Company, Philadelphia, pp. 104-105

[6] Von Neumann J. (1948) The general and logical theory of automata. In cerebral mechanisms in behavior - The Hixon symposium (Jeffress L. A. (ed.), 1951) John Wiley & Sons, Inc., New York, pp. 1-41

[7] Todd, P. M. (1993) Artificial Death, Second European Conference on Artificial Life (ECAL93), pp. 1048-1059.

[8] Sayama, H. (1998) Introduction of Structural Dissolution into Langton's Self-Reproducing Loop, Artificial Life VI, MIT Press, pp. 114-122.

[9] Imanishi, K. (1980) Independence in evolution (Shutai-sei-no-shinka-ron), CHUOKORON-SHINSHA, INC., pp. 218 (in Japanese)

Proposal of a general simulator "SIVA"
for heterogeneous environment and self-decomposable ALife

Tsutomu Oohashi [1], Tadao Maekawa [2], Osamu Ueno [3, 4],
Emi Nishina [1, 5], Norie Kawai [1, 4], and Katsunori Shimohara [1]

[1] ATR Human Information Processing Research Laboratories, JAPAN
[2] ATR Media Integration & Communications Research Laboratories, JAPAN
[3] Department of Biochemistry, Gifu University School of Medicine, JAPAN
[4] Foundation for Advancement of International Science, JAPAN
[5] National Institute of Multimedia Education, JAPAN
Contact: oohashi@hip.atr.co.jp

Keywords: Programmed Self-Decomposition, PSD model, death, ecosystem, computer simulation, evolution

Abstract

We have proposed and verified a theoretical hypothesis of the programmed self-decomposition (PSD) model [1-5] postulating that a self-decomposition mechanism is programmed into all living organisms on earth, and that it contributes to the material as well as the spatial restoration of the ecosystem. In a series of computer simulation studies, we reported that (1) the PSD mechanism contributed to the restoration of a virtual ecosystem; (2) the PSD mechanism advanced evolutionary adaptation of virtual life forms; (3) the virtual mortal life forms based on the PSD model were not exterminated by the immortal ones in almost all cases even when both the mortal and immortal ones started their lives in the same virtual ecosystem having finite space and materials with a heterogeneous distribution; and (4) the PSD mechanism could appear by evolution and survive. In this report, based on this hypothesis and the above findings, we propose a general simulator called "SIVA," which can be widely applied to various investigations of life as well as social activities by any research partners who are interested in the significance of death. A sample implementation

version of the SIVA-Platform and other information will be provided.

1. "Programmed Self-Decomposition" Model
1.1 Restoration Mechanism of the Terrestrial Ecosystem

We suppose that death contributes to the restoration of the ecosystem and to evolutionary advancement and that death itself might be obtained as a fruit of evolution. The terrestrial ecosystem can be characterized as being almost closed, in that space and substance are both limited. Accordingly, to maintain the stability of that terrestrial life form activity, the space and substance that are removed from the environment by the activity of the life form itself have to be returned to the environment. We have set up a new hypothesis that is complementary to that of the food chain to explain this restoration of the environment. In the present terrestrial ecosystem, while such a mechanism of restoring the environment to its original state is occurring, another hidden mechanism is fundamentally built into every living organism by which each cell of that organism positively decomposes itself so as to contribute to the restoration of the environment [1]. We call this phenomenon of decomposing oneself "self-decomposition."

Table 1   Self-Reproduction~Self-Decomposition (SRSD) System

Automaton A	constructs another automaton according to instruction I
Automaton B	makes a copy of instruction I
Mechanism C	combines A and B and functions as follows:
	1) let A construct another automaton according to I
	2) let B make a copy of I and insert it into the automaton constructed above
	3) separate the new automaton from the system A + B + C
Automaton D	consists of A + B + C
Instruction I_D	describes automaton D
Automaton E	consists of D + I_D, which can reproduce itself
Automaton FZ	has the ability to disjoint the whole system into component elements
Instruction I_{D+FZ}	describes automaton D plus automaton FZ
Automaton E_{FZ}	consists of D + I_{D+FZ}, which can reproduce itself and construct another automaton FZ
Automaton G	composed of E_{FZ} and FZ, viz., D + FZ + I_{D+FZ}

1.2 Theoretical Definition of Self-Decomposition

We have developed a concept of a self-reproductive, self-decomposable automaton (Table 1) [1] through a modification of von Neumann's self-reproductive automaton model [7]. The system G described in Table 1 can reproduce itself, and therefore makes FZ a subsystem within the system. FZ has the ability to disjoint G into finite elements. These elements are sized and structured in such a way that the entire ecosystem that G belongs to may take advantage of them collectively. FZ's mode of action can be one of the three shown in Table 2.

Table 2 Three modes of action of decomposing module FZ

(1) Its production is normally restricted.
With the input of a particular message, however, the restriction is lifted and the production activated.
(2) Its operation is normally restricted.
With the input of a particular message, however, the restriction is lifted and the operation activated.
(3) (1) and (2) together.

Furthermore, if, after a certain amount of time has passed or a certain set of events has occurred, there is still no message input to trigger an action, (1), (2), or (3) will happen automatically.

System G composed as described above not only reproduces itself, but also has the ability to put an end to its own life and to return to its origin, that is, to contribute to restoring the ecosystem to its original state. We tentatively call this system a "self-reproductio-self-decomposition (SRSD) system" and call the theoretical model under discussion a "programmed self-decomposition (PSD) model."

1.3 Verification of the PSD Model

We adopted two approaches in order to test the above model: biological experiments using actual life forms and computer simulations using virtual life forms. According to the working hypothesis that the lysosomes in each cell should correspond to FZ [1, 2], we have conducted biological experiments and begun to obtain data suggesting that the mechanism of self-decomposition is actually installed in living organisms (to be reported in the near future). In terms of the computer simulation studies, we have obtained the following results using a series of simulators called "SIVA," which are distinguished from the other ALife simulators by their finite and heterogeneous environmental conditions and by the rational life activities based on the above automaton model, which are designed to resemble the terrestrial ecosystem and life forms, respectively.

Merits of the PSD mechanism

We found that the mortal life forms based on the PSD model had fundamental advantages over the immortal life forms when they lived independently in two similar environments (Figure 1). The mortal life forms based on the PSD model contributed to restoring the simulated ecosystem and continuously reused the limited space and substances. The mechanism also provided greater opportunity for evolutionary adaptation [2].

non-PSD (immortal)   PSD (mortal)
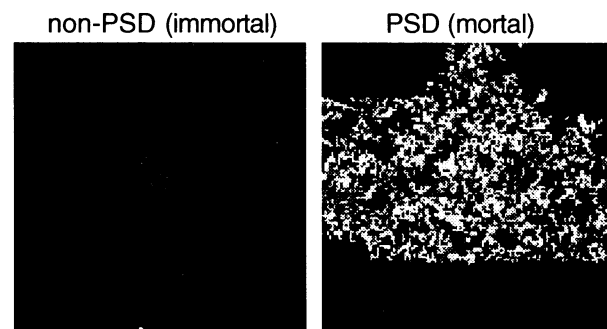
A. Individual distributions after 4000 time count of simulations
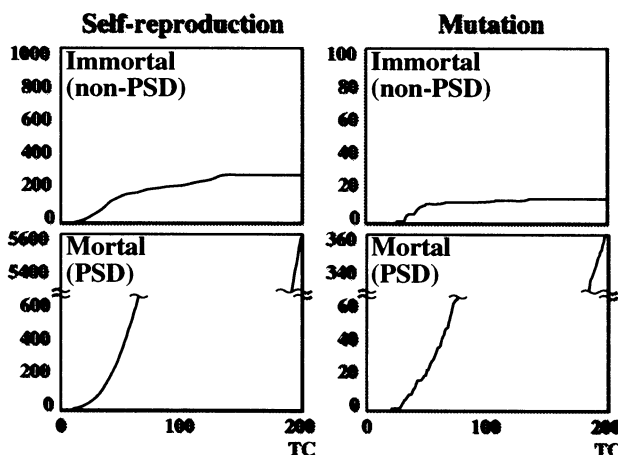
B. Frequency of reproduction and accumulated frequency of mutation

Figure 1. Comparisons between the PSD life forms and the non-PSD life forms [2]

Evolutionary advantage of the PSD life forms

The mortal life forms based on the PSD model were not exterminated in almost all cases, but gained advantages over the immortal life forms in many cases, when both a mortal individual and an immortal one started their lives in the same finite, heterogeneous ecosystem (Figure 2) [3, 4]. When the PSD mechanism evolved from immortal, or more primitive, life forms, the PSD-based mortal life forms were not exterminated and did survive. By the same token, the PSD-based mortal life forms were not exterminated when the immortal life forms appeared evolutionarily by losing the PSD mechanism due to mutation [5].

A. Fixed conditions   B. Diffusible conditions
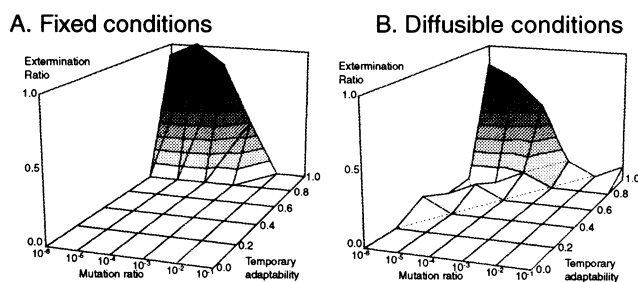
Figure 2   Extermination ratio of the PSD model based mortal life forms

These results suggest that the PSD mechanism which introduces an individual to death should be a fundamental and advantageous strategy for virtual life forms, and that it may have contributed to the evolutionary development of the actual terrestrial ecosystem [5].

2   A General ALife Simulator called "SIVA-Platform"

2.1   Objective

On the basis of the above concept and findings, a general simulator called "SIVA-Platform" has been developed to simulate virtual life activities based on the PSD model in finite and heterogeneous environments. The PSD mechanism, which seems to be a biologically significant strategy, may also exist in various phenomena including life activities and social activities. If its existence were not yet revealed, the installation of the PSD mechanism into the simulations of such phenomena might enable simulations that are more precise. Moreover, by providing a public platform to study the biological significance of death, we expect to find our research partners sharing common ideas. In addition, one can retest our previous studies.

We have tried to install the general characteristics of life activities by referring to scientific knowledge on the actual terrestrial ecosystem or lives. At the same time, we have tried to make the system as simple as possible.

2.2   Characteristics of the SIVA-Platform

The characteristics of the SIVA-Platform are shown in Table 3. The platform has an environment and individuals as its components. Finite and heterogeneous environmental conditions are provided. Simulative individuals are designed as a kind of the SRSD system, and consist of virtual biological molecules. The molecules are classified into Automata, Information, and Energy according to their roles. Automata and Information have a 4-layer system (Polymer, Sub-Polymer, Monomer, and Element). One can design the diversity of the structure and the activities of the individuals in a well-ordered style.

Any activities of the individuals are defined using SIVA-Language, each word of which is equivalent to one Automaton-Sub-Polymer. The individuals can execute not only fundamental self-reproduction and self-decomposition, but also more complex information processing. Each Automaton is synthesized according to Information which can be transmitted and sometimes mutated. Consequently, one can observe hereditary or evolutionary phenomena.

Table 3   Characteristics of the SIVA-Platform

	Characteristics
Environment	Finite (Substance, Energy, Space) Heterogeneous (Substance, Energy)
Individuals	Based on the SRSD system Virtual biological molecules are classified according to their roles in the activities of the individuals - Automaton - Structure Automaton - Function Automaton - Temporary Information Automaton - Information - Constitutive Information - Energy Virtual biological molecules compose a multi-layer system - Polymer - Sub-Polymer - Monomer - Element

2.3   Implementation of the SIVA-Platform

Overview

In this section, we report how the current version of the SIVA-Platform was implemented. Figure 3 shows an overview of the SIVA-Platform.
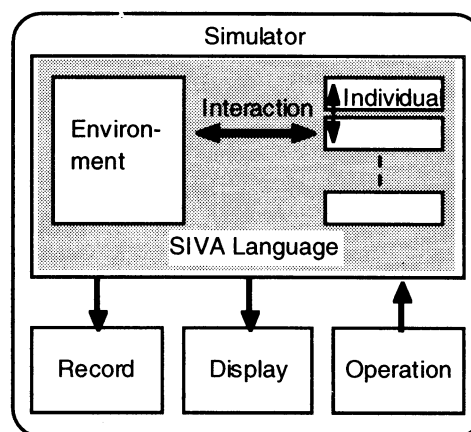
Figure 3   Construction of the SIVA-Platform

Table 4. Construction of the Virtual Bio-molecular System

Role \ Layer	Automaton			Information	Energy
	Structure	Function	Temporary Information	Constitutive Information	
Polymer	↑Array	↑Array	↑Array	↑Array	
Sub-Polymer	5 Characters of Monomer	5 Characters of Monomer	5 Characters of Monomer	5 Characters of Monomer	
Monomer	↑Array STUV	↑Array OPQR	↑Array KLMN 0~9	Translation ↑Array WXYZ	J
Element	Synthesis↑ ABCD(FGHI) Synthesis↑				E

Table 5 Positions of the Virtual Biomolecular Automata in SIVA-Language

Role \ Layer	Automaton
Polymer	Sentence / Set of them
Sub-Polymer	Word
Monomer	Character
Element	Data

Programming Language

JAVA 1.2.2 is used as the programming language to make the SIVA-Platform available on many computer environments.

Algorithm

Simulations are conducted according to the flow charts shown in Figure 4. After the initialization process, the activities of each individual are simulated in pseudo-parallel by 1 time count (TC) until the user-defined TC. The activities of the individuals are defined by using SIVA-Language (see below). The diffusion of the environmental substances and the reproduction of energy in the environment can be defined before the subsequent TC.
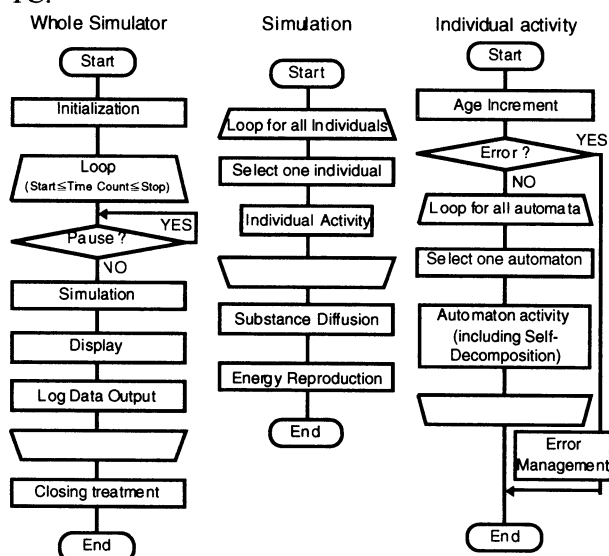
Figure 4   Main flow charts of the SIVA-Platform

Environment

The virtual space of the SIVA-Platform is a finite 2-dimensional lattice according to the user's configuration. The initial values of physical conditions, such as the amount of substances, amount of energy, and temperature, can be defined for each block of the lattice independently. Each block consists of latticelike pixels according to the user's configuration. Each pixel can be occupied by one individual. Environmental conditions can be defined with a plain text file.

Individuals

The SRSD system has been reorganized in consideration of the bio-system in general life forms. The virtual individuals consist of virtual biological molecules which are categorized as shown in Table 4 according to their roles and layers. The alphabetical characters in the table denote the biological molecules defined in the present sample implementation. Table 5 shows how each category of the virtual bio-molecular Automata are positioned in SIVA-Language. All of the virtual bio-molecular Information and Energy are positioned as data in SIVA-Language.

Role   Automata correspond to the phenotype of the actual life forms. Automata can be classified into three types: structure, function and temporary (not constitutive) information processing. Constitutive Information corresponds to the genotype of the actual life forms. According to the sequences of Constitutive Information, each Automaton is synthesized.

Layer   Each Monomer of Automaton, Information, and Energy is synthesized from Elements according to the formation rules defined in formation-rule-tables (not shown). Each set of 5 characters of Automaton-Monomer and Information-Monomer functions as a Sub-Polymer. One Constitutive Information-Sub-Polymer decides which Automaton-Monomer was synthesized. One Automaton-Sub-Polymer corresponds to one word of SIVA-Language (Table 5 and 6).

A sequence or sequences of Automata-Sub-Polymers are called Automaton-Polymer. One Automaton-Polymer corresponds to a sentence or a set of sentences of SIVA-Language and decides the activities of the individual. An example of the SIVA-Language representation of an activity of an individual based on the above components is shown in Table 7. The current version of SIVA-Language can include more than 3000 words, which enable it to design various kinds of life activities. However, additional commands or variables for the SIVA-Language require notation of the new function using JAVA.

The above briefly mentioned rules for individuals can be arranged using individual-definition files, formation-rule-table files, and translation-rule-table files written in plain text. Formation-rule-tables can define the necessary amount of energy for the combinations.

We will provide not only a sample implementation version of executables, but also the source code and other information so that users can modify or reconstruct the platform if necessary.

3. Conclusion

Our previous studies suggest that the PSD mechanism introducing an individual to death is a fundamental and advantageous strategy for virtual life activities. A general simulator called "SIVA-Platform" has been developed to simulate virtual life activities based on the PSD model in a finite and heterogeneous environment. The SIVA-Platform can be applied to various investigations of life as well as social activities by any research partners who are interested in the significance of death.

Acknowledgment

The authors would like to gratefully thank Mr. Masatoshi HIROSE, Ms. Hisako HAYAKAWA, and Mr.

References

[1] Oohashi T., Nakata D., Kikuta T. and Murakami K. (1987) Programmed self-decomposition model, Kagakukisoron kenkyuu, Vol. 18, No. 2, pp. 79-87 (in Japanese)
[2] Oohashi T., Sayama H., Ueno O. and Maekawa T. (1996) Artificial life based on programmed self-decomposition model, ATR Technical Report TR-H-198
[3] Oohashi T., Maekawa T., Ueno O., Nishina E. and Kawai N. (1999) Requirements for immortal ALife to exterminate mortal ALife in one finite, heterogeneous ecosystem, 5th European Conference on Artificial Life (ECAL99), pp. 49-53.
[4] Maekawa, T., Ueno, O., Nishina, E., Kawai, N., Shimohara, K., and Oohashi, T. (2000) Diffusible immortal ALife rarely exterminate diffusible mortal ALife in one finite, heterogeneous ecosystem, the 5th international symposium on Artificial Life and Robotics (AROB2000), in this proceedings.
[5] Maekawa, T., Ueno, O., Nishina, E., Kawai, N., Shimohara, K., and Oohashi, T. (2000) Evolutionary advantage of Self-Decomposition mechanism, the 5th international symposium on Artificial Life and Robotics (AROB2000), in this proceedings.
[6] Odum E. P. (1971) Fundamentals of Ecology, Third ed., W. B. Saunders Company, Philadelphia, pp. 104-105
[7] Von Neumann J. (1948) The general and logical theory of automata. In cerebral mechanisms in behavior - The Hixon symposium (Jeffress L. A. (ed.), 1951) John Wiley & Sons, Inc., New York, pp. 1-41

Table 6   Formation and Functions of Automata-Sub-Polymers (extracted)

Function-Sub-Polymer			Temporary Information-Sub-Polymer								
Reproduction/Decomposition			Operator			Variable			Value		
monomers	word	meanings	monomers	word	meanings	monomers	word	meanings	monomers	word	meanings
OPxxx	syntha	Synthesis	KKKxx	=	equal	LKKKx	age	age	00000	0	0
POxxx	decam	Decomposition	KLKxx	>	more than	LKKLx	unconf	degree of unconformity	99999	99999	99999

* x denotes any character.

Table 7   SIVA-Language representing a Life Activity (example)

ID4 CR	← ID of Automaton-Polymer
unconf > 2 decam decim CR	← If unconf exceeds 2, then self-decomposition

Hybrid Force and Position Control of Miniature Robotic Fingers Driven by Piezoelectric Actuators

Zhongwei Jiang[*1], Seiji Chonan[*2] and Yoshihiro Namiki[*2]
1 Department of Mechanical Engineering
Yamaguchi University, Ube, 755-8611, Japan
Email:jiang@po.cc.Yamaguchi-u.ac.jp
2 Department of Mechatronics and Precision Engineering
Tohoku University, Sendai, 980-8579, Japan

Abstract

This paper is concerned with a study on a pair of two miniature flexible fingers driven by piezoelectric actuators and stepping motor for grasping an object with prescribed minute force and motion. The artificial finger is a uniform flexible cantilever equipped with a bimorph piezoelectric actuator at its base and a force sensor at the tip. The two-fingered gripper is first driven by the stepping motor until its tip touched the object, then the piezoactuator is switched on to control the minute grasping force and position. The theoretical model of the electro-mechanical system is developed and H infinity controller is designed based on the theoretical simulation for a control problem where the gripper is commanded to hold an object with a time-constant force at 0.01N performing its prescribed task. The theoretical and experimental results showed that the present piezo-actuated gripper has good control behaviors for sophisticated grasping work.

1 Introduction

Advances in technology have accelerated the development of robotic hands having varied uses and serving many functions. Particularly in the field of biotechnology, the demand on soft handling grippers that hold small, lightweight, or fragile objects have significantly been enhanced.

Artificial hands and fingers capable of grasping a variety of objects have been studied by several researchers [1-4]. Most of the robotic fingers designed so far have been actuated by servomotors, stepping motors or linear ball bushings, for which the effects of compliance, backlash and hysteresis loss cannot sometimes be ignored. Okamoto et al [5] developed a two-finger gripper driven by shape memory alloy (SMA) wires. The SMA works well as an actuator to do tasks at low speed. However, this is not the case for high-speed tasks. One of the actuators that covers this point is the piezoelectric ceramic, whose deformation is proportional to the applied voltage and the functional motion is reversible for both the positive and negative driving signals.

This paper presents a theoretical and experimental study on the hybrid position and force control of a parallel two-fingered miniature gripper. The fingers are flexible cantilevers actuated by piezoelectric bimorph strips at the base and are supported by linear ball bushings which ride on a steel shaft. One finger is equipped with a compact grasping force sensor at the tip. The system control characteristics are examined on the problem where the gripper is commanded to grasp defferent objects with a time-constant force 0.01N performing a prescribed, commanded motion.

2 Formulation of Problem

Figure 1 illustrates a two-fingered parallel miniature gripper equipped with a compact force sensor at the tip of one finger. Each finger consists of a piezoelectric ceramic bimorph at the base and a flexible copper cantilever at the tip. The cantilever is driven laterally by the bending deformation of the ceramic actuator. The maximum displacement of the actuator is approximately 300 μm. Accordingly, the resulting fingertip displacement is not sufficient for grasping a moderately large object. To compensate for the small displacement of the fingers, they are supported separately by linear ball bushings which ride on a fine screw thread (SKS FKB-D401A). The bushings are driven by a step servomotor in the opposite direction. One pulsation signal to the motor opens/closes the gripper about 5 μm. The linear slide mechanism takes charge of the coarse but slow motion, and the piezo-actuated cantilever takes charge of the fast and fine motion of
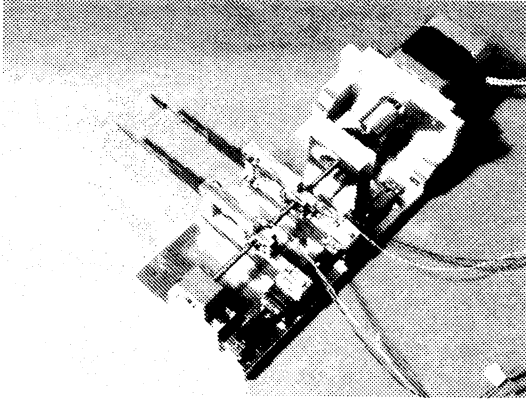
Fig.1. Two-fingered parallel gripper.
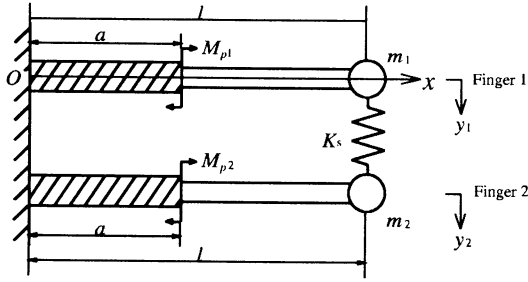
Fig.2. Geometry of problem and coordinates.

the gripper.

Figure 2 illustrates the mathematical model of the parallel two-fingered system. A compact grasping force sensor is attached to the tip of Finger 1. In the theoretical analysis, the fingers are described respectively by a stepped beam of rectangular cross-section with a change of cross-section at $x = a$. K_s is the total stiffness of the sensor and the grasped object. The mass of the sensor and that of the object are distributed onto the tips of the fingers, which are represented respectively by m_1 and m_2 in the figure. The mass per unit length of the Finger i (=1,2) is denoted by $\rho A_I(x)$ and the flexural rigidity by $EI_I(x)$, both of which are given by

$$\rho A_i(x) = \begin{cases} \rho_b^i A_b^i & \text{for } a \leq x \leq l \\ \rho_p^i A_p^i & \text{for } 0 \leq x \leq a \end{cases}$$

$$EI_i(x) = \begin{cases} E_b^i I_b^i & \text{for } a \leq x \leq l, \\ E_p^i I_p^i & \text{for } 0 \leq x \leq a, \end{cases}$$

$$i = 1, 2. \qquad (1)$$

Here, ρ, A, E, and I are the mass density, cross-sectional area, Young's modulus and moment of inertia of the finger; the superscript i refers to Finger i and the subscripts b and p the beam and the piezoactua-

Table 1. Physical parameters of two-fingered system.

Property	Beam	Actuator
Length (mm)	36.2	31.8
Width (mm)	6.0	12.0
Thickness (mm)	0.3	0.65
Mass Density (kg/m^3)	8670	8300
Young's modulus (GN/m^2)	102.5	58.03
Damping Coefficient (s)	3.5587×10^{-5}
Tip mass (kg)	0.585×10^{-3}
Sensor mass (kg)	0.295×10^{-3}
Sensor stiffness (N/m)	9.542×10^3
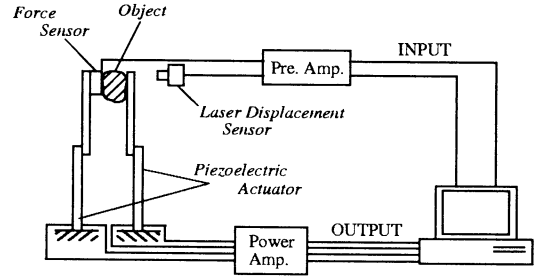
Fig.3. Experimental setup.

tor. In following, it is assumed that the flexible fingers deform only in the xy plane and the force sensor at the fingertip is in contact with the handled object. The equations of motion governing the flexural vibrations of fingers [6] are

$$[\rho A_i(x) + m_i \delta(x - l)] \frac{\partial^2 y_i}{\partial t^2}$$
$$+ \frac{\partial^2}{\partial x^2} \left[EI_i(x)(1 + \gamma_i \frac{\partial}{\partial t}) \frac{\partial^2 y_i}{\partial x^2} \right] \qquad (2)$$
$$= M_{pi}[\delta'(x) - \delta'(x - a)] - F_{si}, \quad i = 1, 2$$

where y_i is the deflection of the finger, γ_i is the internal damping coefficient, and M_{pi} is the continuous constant bending moment from $x = 0$ to a induced in the piezoelectric actuator; $\delta(\cdot)$ is the Dirac delta function, and $\delta'(\cdot) = d\delta/dx$; F_{si} is the fingertip grasping force measured by the sensor and is represented theoretically by

$$F_{s1} = -F_{s2} = K_s[y_1(x,t) - y_2(x,t)]\delta(x - l). \quad (3)$$

The bending moments M_{pi} are regulated by a digital H_∞ controller.
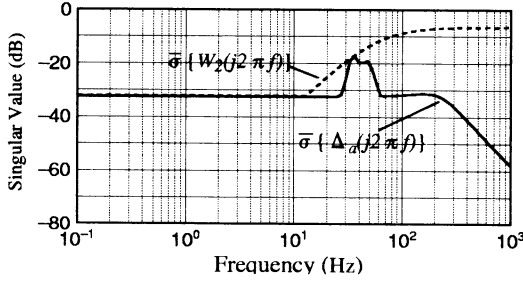
Fig.4. Plot of $\Delta_a(s)$(solid) and $W_2(s)$(dash).
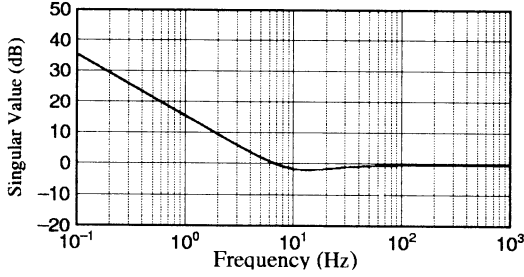
Fig.5. Plot of $W_1(s)$.

3 Results and Discussion

The demensions and properties of the gripper are presented in Table 1 and the overall experimetal setup in Finger 3.

The gripper is designed for the purpose that the object is handled at a time-constant minute force in operation of a prescribed motion, further, the grasping tasks should be done in stable without influence of disturbances such as the change of object in mass, size and so on. Therefor, the H-infinity algorithm, which is good for robust performance, is introduced to control the bending moments M_{p1} and M_{p2} induced by piezoactuators. In controller design, Finger 1 is considered to be controlled by using only the force sensor measurement and Finger 2 to be driven by employing its own endpoint displacement as feedback signals.

Suppose that the normal plant transfer function is $G_m(s)$ and consider perturbed plant transfer function $G(s)$ with an unstructured additive perturbation $\Delta_a(s) = G(s) - G_m(s)$, then $\Delta_a(s)$ can be wriiten as

$$\Delta_a(j\omega) = U \begin{bmatrix} \sigma_{a1}(j\omega) & O \\ O & \sigma_{a2}(j\omega) \end{bmatrix} V^H \qquad (4)$$

where U and V are unitary. For robust performance the weighting function $W_2(s)$ of the form

$$W_2(j\omega) = \begin{bmatrix} W_{21}(j\omega) & O \\ O & W_{22}(j\omega) \end{bmatrix} \qquad (5)$$
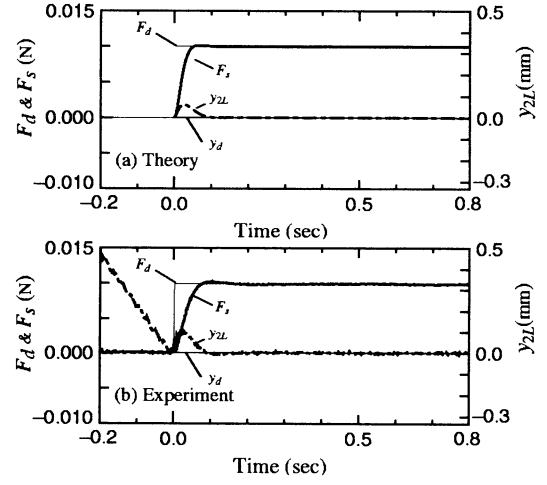
Fig.6. Time response of gripper grasping a normal object at a constant force 0.01N with H_∞ controller. F_s is the sensor output of grasping force, F_d is demanded grasping force, y_{2L} is the tip displacement of Finger 2 and y_d is the prescribed position.
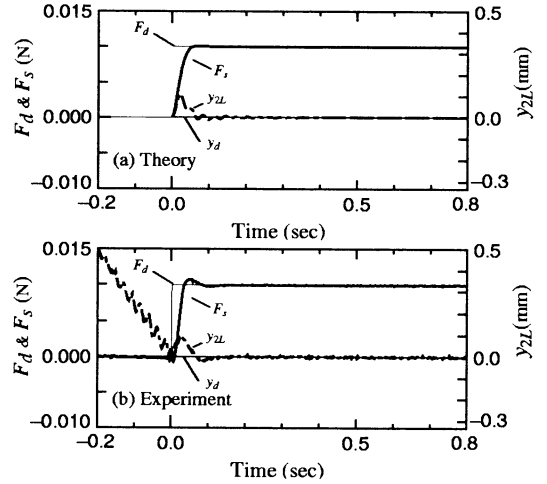
Fig.7. Demonstration of grasping performance for an object having 2.5 times mass perturbation with same H_∞ controller of Fig.6.

is introduced to satisfy the condition

$$\bar{\sigma}\{\Delta_a(j\omega)\} < \bar{\sigma}\{W_2(j\omega)\} \qquad (-\infty < \omega < \infty)(6)$$

The gains of the selected weighting function $W_2(s)$ and transfer function additive perturbation in the endpoint mass $\Delta_a(s)$ are shown in Fig.4. Further, a coordinate weighting function $W_1(s)$ for sensitive transfer function of the plant is tuned as shown in Fig.5.

The gripper's behaviour are tested on the problems where the gripper is commanded to hold the object at
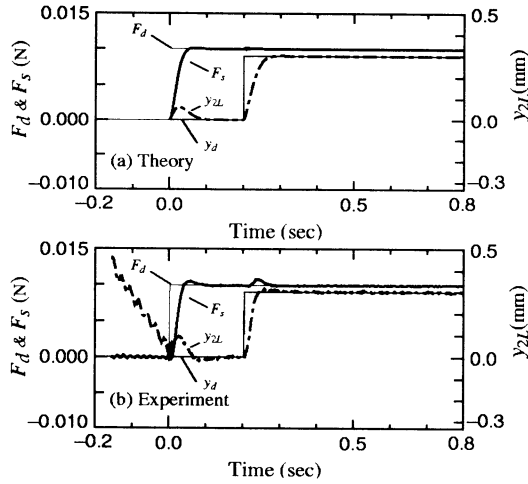
Fig.8. The object is operated following a trace of step function displacement y_d=0.3mm with a constant force 0.01N after it is kept for 0.2sec at the initial position.

a time-constant force 0.01N on several prescribed motions. Figure 6 shows theoretical and exmerimental results when the girpper is ordered to grasp an object of 0.6g. The gripper system works in such a way that the stepping motor is first driven until the tip of Finger 1 touching on the object, then the piezo-actuators are switched on to regulate the object for a prescribed task. Figure 7 demonstrates the performance when the gripper operates an object whos mass is 2.5 times bigger than the one in Fig.6. Figures 8 and 9 demonstrate the performance when the object is held in a constant force (0.01N) and then is operated in step or sinusoidal function motion. All above resutls indecate that the piezo-actuated miniature girpper works well for grasping a minute object at a prescribed task. Furthermore, in each figures, Fig.(a) illustrates the theoretical calculations for determination of the parameters of H_∞ controller and (b) the corresponding experimental results obtained by the theoretically designed controller. The control behaviours both for theoretical simulation and experimental test are compared well with each other, hence it leads to an understanding that the miniature gripper can be designed and tuned systematically with the aid of the computer simulation.

4   Conclusions

A theoretical and experimental study has been presented for operating a two-fingered gripper to hold an object with a time-constant force on a prescribed task. To design the H_∞ controller for robust performance
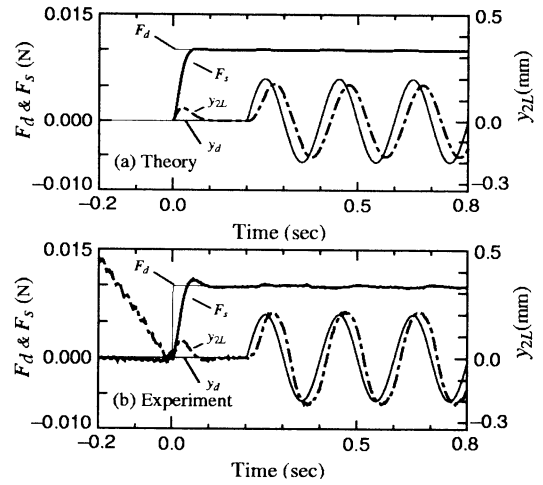
Fig.9. The object is operated following a trace of sinusoidal function displacement $y_d = 0.2 \sin 2\pi 5(t - 0.2)$ mm with a constant force 0.01N.

the theoretical model of a parallel piezo-actuated fingers was derived and was validated in experiment. It showed that the controller parameters could be determined by the theoretical simulations. Furthermore, the obtained results indecate that the piezoactuator is efficient for the hybrid position and force control of a miniature two-fingered gripper holding a delicate object.

References

[1] J.K.Salisbury and J.J.Craig "Articulated Hands: Force Control and Kinematic Issue," *Int. J. Robotic Res.*, Vol.1, pp.4–7, 1982.

[2] P.Dario and G.Buttazzo, "An Anthropomorphic Robot Finger for Investigating Artifical Tactile Perception," *Int. J. Robtic Res.*, Vol.6, pp.25–48, 1987.

[3] H. Terasaki and T.Hasegawa, "Motion Planning for Intelligent Manipulations by Sliding Operations with Parrallel Tow-Fingered Grippers," *J. Robotics Soc. Japan*, Vol.12, pp.138-147, 1994.

[4] J.P.Baartman and T.Storm, "Flexible Grippers for Mechanical Assembly," *Industrial Robot*, Vol.21, pp.23-27, 1994.

[5] T.Okamoto, O.Kitani and T.Torii, "Robotic Transplating of Orchid Protocorm in Mericlone Culture," *J. Soc. Agricultural Machinery, Japan*, Vol55., pp.103-110, 1993.

[6] Z.W.Jiang, S.Chonan and J.Tani, "Tracking Control of a Miniature Flexible Arm Using Piezoelectric Bimorph Cells," *Int. J. Robotics Res.*, Vol.11, pp.260-267, 1992.

Development of a Miniature Spray Robot
Using Conduit Guided Wire for a Die Casting Process

Masaru Hisano,Yasutaka Takeuchi,

Katsutoshi Kuribayashi,Tatsuo Sakamoto

and Kousei Murakami

Faculty of Engineering Yamaguchi University

2-16-1 tokiwadai, Ube, Yamaguchi 755-8611, Japan

Abstract

This paper presents the development of a miniature spray robot using conduit-guided wire for a die casting process. The design, kinematics and spray control planning of the robot are described.

Key words:Die Cast, Spray, Serial link mechanism

1 Introduction

Recently, a die casting process is used as the manufacturing process of a precise component such as car and camera components. In the process, spray system of water with some powder to the die is required to be automated. In our study, a new automatic spray system using robots is proposed.

This spray robot system is consisted of two robots that are called the main robot and the wrist robot[3]. The wrist robot has a spray nozzle, and changes its pose and position for the die. Conventional spray wrist robot is too large, because motors are assembled in the robot. To solve this problem, the wrist robot is divided into motor part and robot link part who are connected with conduit guided wires to conduct power[1][2]. In the presentation, we report 1) the designed mechanism, 2)the inverse kinematics and 3)the simulation results for the proposed spray robot system.

2 Design of the robot system

In actual die spray operation, the wrist robot must be entered narrow space between two dies in the die casting machine as shown in Fig.1. According to this condition, it is necessary for this robot to have three degrees of freedom

(θ_x, θ_y, z). Also, this robot is needed to have the following specifications.

1. The range of the rotational angle is $\pm 45°$.
2. The maximum rotational velocity is $90°/s$.
3. The target robot size is 100mm×100mm×100mm.
4. The maximum weight is about 2.0kgw.

To make small spray robot, this robot is divided into motor parts and robot link parts who are connected with the conduit guided wires to conduct power. Next, the conduit-guided wires, and the main and the wrist robot will be explained.
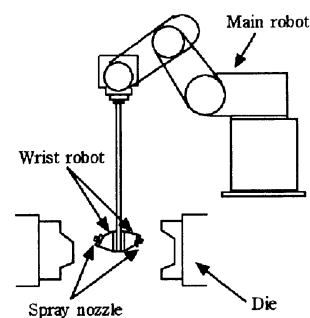
Fig.1 Spray robot layout

2.1 The conduit-guided wire
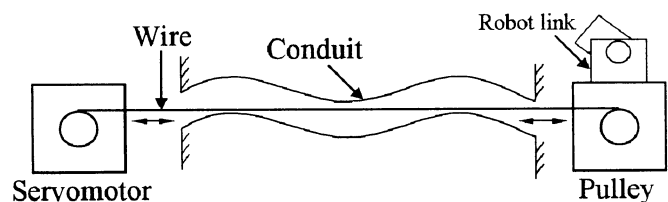
Fig.2 The conduit-guided wire system

Fig.2 shows the principle of the conduit-guided wire system. One end of the wire is joined on the motor, the other end is joined on the pulley. Pulling the wire in the conduit-guided can conduct power. However the friction of the conduit-guided wire must be considered[4].

2.2 Requirement for the main and the wrist robot

The main robot The main robot is to be equipped with wrist robot to spray. This robot has three degrees of freedom, and is requseted only to change the position for the die, not to change the pose.

The wrist robot The spray nozzle is attached on the wrist robot. This spray nozzle must be kept perpendicular to the die surface. This robot has three degrees of freedom : one translational and two rotational axis. Also, the wrist robot's speed must be faster than the main robot's speed.

2.3 Design of the wrist robot
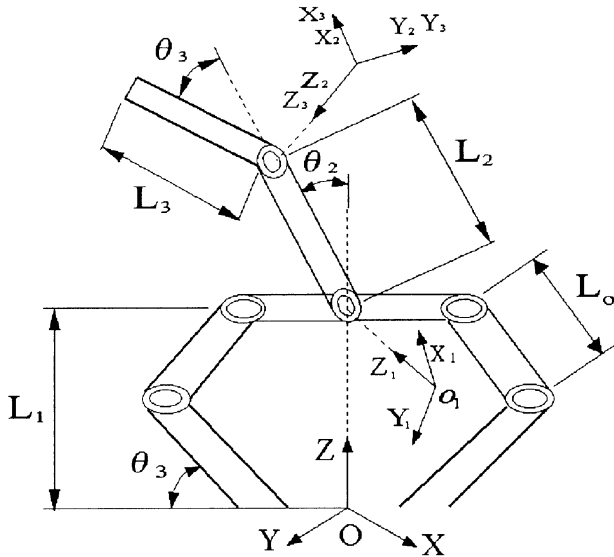
Fig.3 The coordinate system of the wrist robot

This wrist robot with a spray nozzle attached to a main robot is consisted of two parts. One part is the serial link part with a spray nozzle to perform actual spray work. The other part is the servomotor part. Two parts are joined with the conduit-guided wire, which is fixed on the each pulley. By Pulling the wire the link which joined on the pulley is leaned . This serial link part is the tip part of the wrist robot. Furthermore, by using the parallel link mechanism the wrist robot can move to the z-axis as shown in Fig.3. Fig.4 shows the wrist robot layout.
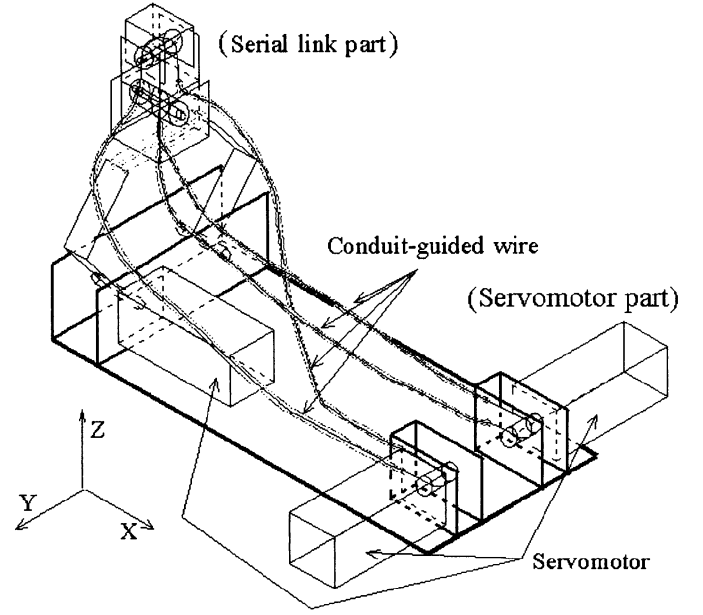
Fig.4 The wrist robot layout

3 Kinematics

In this chapter, in order to control this wrist robot, the kinematics of the wrist robot is described. It is assumed that the coordinate system is fixed for the wrist robot as shown in Fig.3. L_1, L_2 are the length of serial link part, L_3, L_o are the length of parallel link part. Also, θ_1, θ_2, θ_3 are the angles of each motor. The kinematics are given by

$$x = L_3 \sin \theta_2 \qquad (1)$$
$$y = L_3 \sin \theta_1 \cos \theta_2 + L_2 \sin \theta_1 \qquad (2)$$
$$z = L_1 + L_2 \cos \theta_1 + L_3 \cos \theta_1 \cos \theta_2 \qquad (3)$$

where (x, y, z) is the position. Therefore, using (1), (2) and (3), the desired reference rotational angle of the wrist robot can be given by

$$\theta_1 = \arcsin \frac{y}{L_2 + \sqrt{L_3^2 - x^2}} \qquad (4)$$
$$\theta_2 = \arcsin \frac{x}{L_3} \qquad (5)$$
$$\theta_3 = \arcsin\{(z - (L_2 + \sqrt{L_3 - x^2})$$
$$\times \sqrt{1 - \frac{y^2}{(L_2 + \sqrt{L_3^2 - x^2}}})/(2L_o)\} \qquad (6)$$

4 Robot servo control system

The block diagram of a feedback control system for the spray robot is shown in Fig.5, in which P is a feedback controller and K is kinematics of the wrist robot. Also,

θ_{xr}, θ_{yr}, zr are the references for θ_x, θ_y, z. θ_x is rotational angle about the x-axis, and θ_y is rotational angle about the y-axis. θ_{1r}, θ_{2r}, θ_{3r} are the references for θ_1, θ_2, θ_3 and u_1, u_2, u_3 are input signal. The spray robot with a feedback control system can move smoothly. The pulses of rotary encoders of three motors are sent to a personal computer through the counter board. Using these data the velocity is calculated, and by integrating velocity the angle of motor is given every each sampling time. By calculating difference value between actual angle and reference angle the error data is given. The error data are fed back to the controller and an input voltage for each servomotor is obtained.
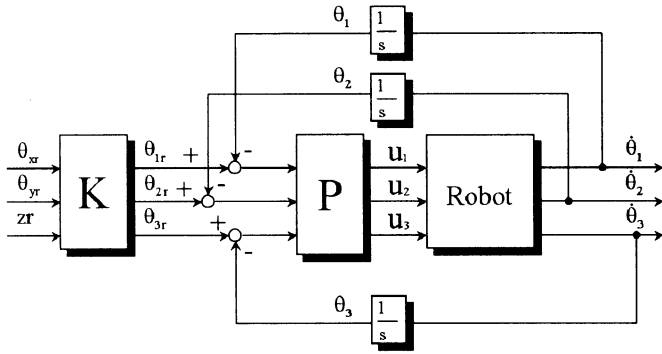
Fig.5 The feedback control system

4.1 Experimental result for single axis

Experiment for single axis was performed. Reference rotational velocity is 1.57 [rad/s]. An experimental time is 5 [sec], and the control sampling period is 5 [msec]. The servo gain is set as K = 10.0. Fig.6 shows the response result of the rotational angle.
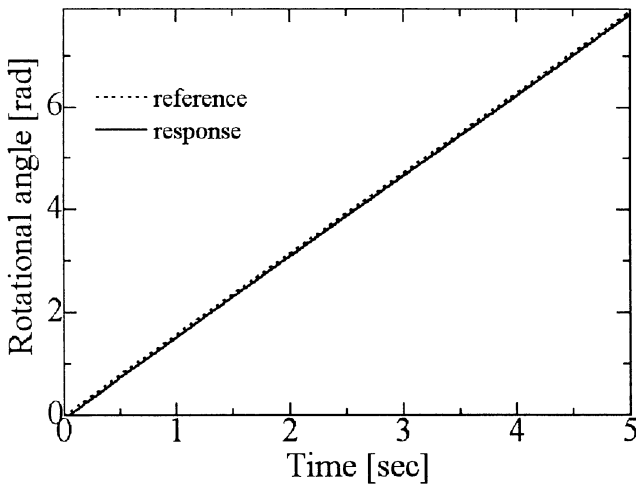
Fig.6 A feedback experiment result of a servo motor

5 Spray control planning

5.1 Spray control system
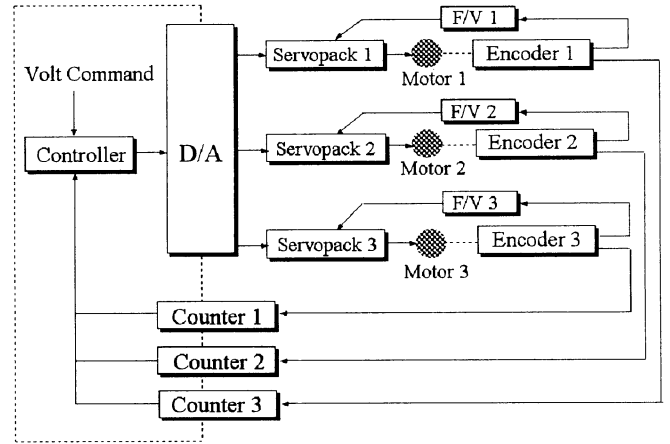
Fig. 7 Spray control system

The control system for the wrist robot is shown in Fig.7. The computer hardware is composed of D/A (PPAT6403) and counter board (CNT24-4). The wrist robot is driven by three servomotors (SGMM-A1C3J26 Yasukawa), where each of them has an optical rotary encoder with 2048 [Pulse/Rev]. The data from the encoder are fed back to the computer through the counter board.

Simple die model shown in Fig.8 was used for the simulation. The range of this die model surface is divided into several mesh points. This robot's pose required is indicated by this figure's arrow.
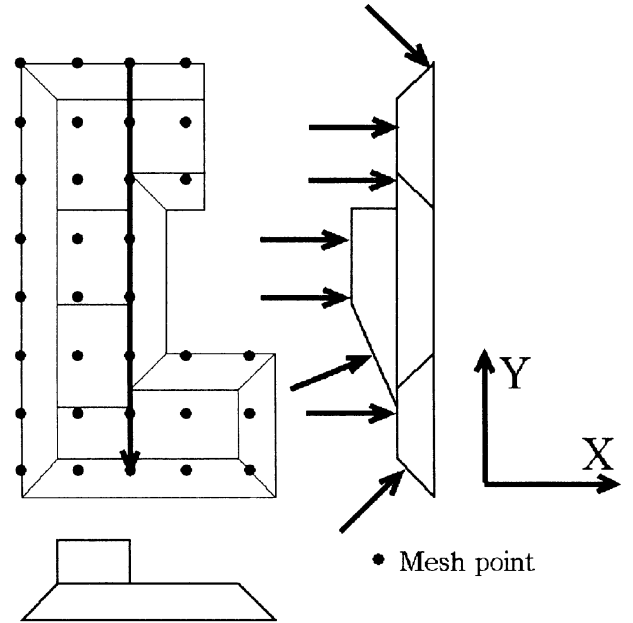
Fig.8 Model die and the pose of the wrist robot

5.2 Simulation of spray trajectory control

A computer simulation was performed to show that the spray trajectory control system is useful. In this simulation, an experimental time is 5[sec], and the control sampling period is 5[msec]. The servo gain is set as K=10.0 for the simulation. This robot changes the position and the pose every one second. Desired rotational angles and positions at mesh points for the die are shown in Table 1.

Table1 Coordinates of the robot at mesh points

Time[sec]	θ_{xr} [$\circ$]	θ_{yr} [$\circ$]	zr [mm]
0	0.00	0.00	0.00
1	45.00	0.00	0.00
2	0.00	0.00	25.00
3	0.00	-45.00	25.00
4	0.00	-90.00	55.00
5	0.00	-90.00	55.00
6	-22.50	-90.00	40.00
7	0.00	0.00	25.00
8	-45.00	0.00	20.00

Spray trajectory control results are illustrated in Fig.9, Fig.10 and Fig.11. In these figures, the horizontal axis is time [sec], the vertical axis is the angles [rad]. From these figures, it is found that this wrist robot can achieve a motion of the references trajectory.

6 Conclusions

In this paper, we have described the design of a spray robot using the conduit-guided wires. The simulation results are produced so that the wrist robot can give the desirable trajectory motion for the die.
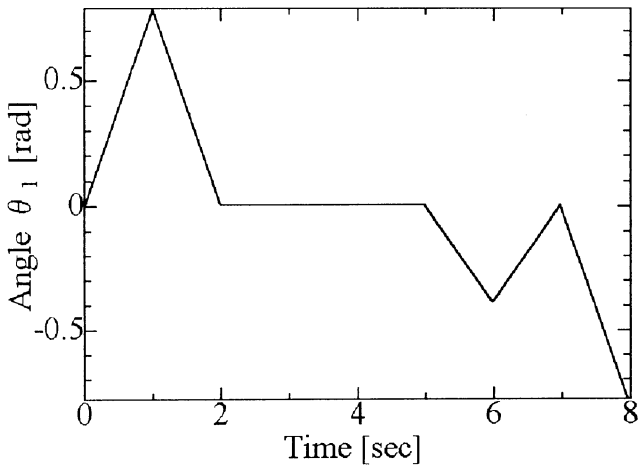
Fig.9 Spray trajectory of θ_1
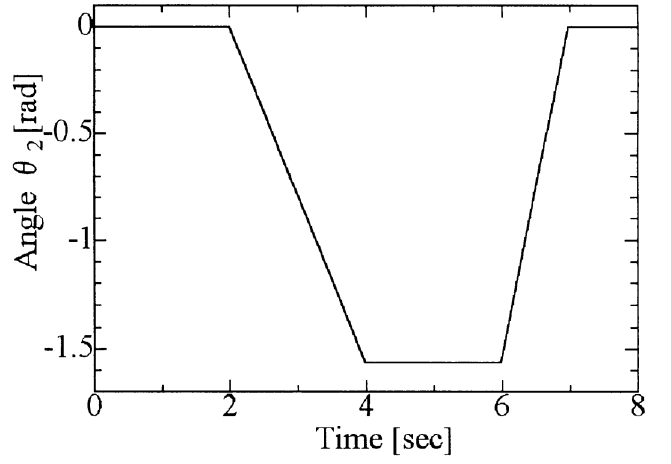
Fig.10 Spray trajectory of θ_2
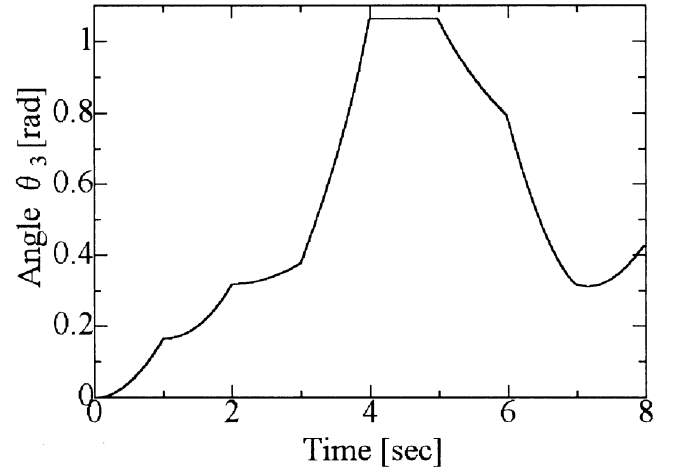
Fig.11 Spray trajectory of θ_3

References

[1] Sigeki S, Hiroyuki S, (1988), Motion Control of the Robot Joint Driven by Wire System(in Japanese).Proceedings of the Annual Conference of the Robotics Society of Japan:59-60

[2] Tokuji O(1978), Force Control of an Artificial Finger Driven by a Hose-Guided Wire(in Japanese). 14-2:46-56

[3] Tatshuo S, Katsutoshi K, Susumu M, Marcelo G. Franco, Jun T, Kousei M,Hideo K(1998), Automatic Path Planning of Spray Robot for a Die Casting Process by Using Genetic Algorithm. AROB 3rd'98 vol.2:448-453

[4] Makoto K(1991), Tendon-Sheath Transmission System and Force Control Issues(in Japanese). Journal of the Robotics Society of Japan vol.9-4:112-115

A Navigation System for Bending Profile of Endoscope Using Strain Gauges

Tetsuya TARUMOTO, Katsutoshi KURIBAYASHI

Faculty of Engineering, Yamaguchi University
2-16-1, Tokiwadai, Ube, Yamaguchi, 755-8611, Japan

Abstract

In this paper, a navigation system for the bending profile of the endoscope using strain gauges in real time is proposed. From the simulations and experiments of 2D, the navigation system has high potential for detecting the bending profile of the endoscope.

key words: Endoscope, Inspection, Navigation, Strain gauge, Large intestine

1 Introduction

An endoscope is used in the medical field for the merits of cutless operation. There are a flexible endoscope and a non-flexible one. The flexible endoscope in the medical field is useful for the inspection of the large intestine. A doctor who is inspecting the large intestine has no information about the bending profile of the endoscope except for X-ray pictures which is seldom taken for the inspection. So, sometimes accidents may occur and the large intestine is broken by the endoscope. In order to avoid this accident, it is necessary to develop a navigation system for the endoscope that gives the doctor the information of bending profile of it in real time. Yamashita have reported the navigation system of the endoscope for paranasal sinus surgery[1]. However, this is the navigator system for the endoscope of non-flexible. The navigation system for the flexible endoscope has not been developed yet. In this paper, a navigation system for the bending profile of the flexible endoscope using strain sensor in real time is proposed.

The strain gages were pasted on the surface of the endoscope and the surface strain of each point were detected. The curvature values of these points were calculated from the strain values. Then the whole bending profile of the endoscope was calculated approximately as a polynomial function of the position along the longitudinal direction of the endoscope based on the Weierstrass's approximation theorem[2].

From the simulations and experiments, the navigation system has high potential for detecting the bending profile of the endoscope.

2 Basic theory

Basic theory of the detection of curvature data on an endoscope and the calculation method of the profile of the endoscope are described as follows. The principle of the detection method of the strain on the surface of the endoscope is shown in Fig.1.
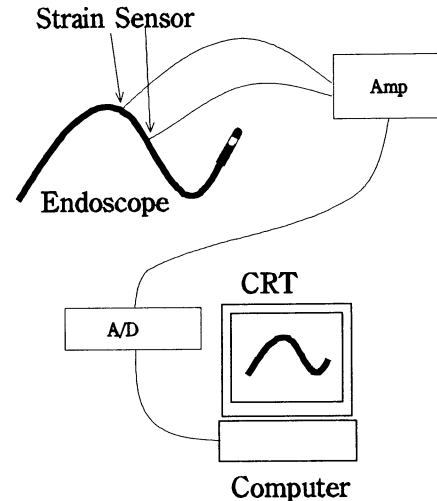
Fig.1 The whole view of the navigation system

2-1 Distance between strain sensors of the endoscope

Strain gauges are used as strain sensors. We must decide the interval distance between gauges and the number of gauges.
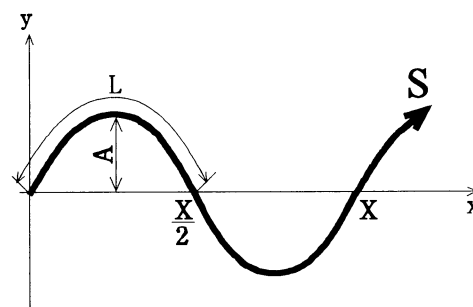
Fig.2 The endoscope of sinusoidal wave

An endoscope of 2D is on x-y axis. On the other hand , s axis is defined as the coordinate along the longitudinal direction of the endoscope.

In this section, in order to decide the interval distance between gauges, it is assumed that the function of the profile of the endoscope is expressed as sinusoidal wave is

$$y = A\sin(\frac{2\pi}{X_t}x) \tag{1}$$

where A[m] is amplitude, $2\pi/X_t$[rad/m] is angular frequency, X_t[m] is a periodic distance. Curvature ρ which is related with the curvature radius is expressed as follows.

$$\rho(x)=\frac{1}{r}=A\left(\frac{2\pi}{X_t}\right)^2\sin\left(\left(\frac{2\pi}{X_t}\right)x\right) \tag{2}$$

where r[m] is curvature radius of endoscope of a certain point along the longitudinal direction of the endoscope.

When the curvature is maximum at eq.(2),

$$\sin\left(\left(\frac{2\pi}{X_t}\right)x\right)=1$$

Then eq.(2) becomes as follows,

$$\frac{1}{r_m}=A\left(\frac{2\pi}{X_t}\right)^2 \tag{3}$$

$$\therefore X_t=2\pi\sqrt{Ar_m} \tag{4}$$

where r_m is the minimum radius of the endoscope. The distance of sampling strain should be less than half of wavelength X_t according to Shannon's sampling theorem[3]. So, the distance between gauges along the x axis should be as follows

$$X_s<\frac{X_t}{2} \tag{5}$$

where X_s is the sampling interval distance for gauges between sensor on the endoscope along the x axis. Substituting eq.(4) into eq.(5), we obtain

$$X_s<\pi\sqrt{Ar_m} \tag{6}$$

Considering eq.(6), the distance L between sensor on s axis should satisfy the following in equality.

$$L<\frac{\min}{X_0}\int_{X_0}^{X_0+X_s}\sqrt{1+\left(\frac{dx}{dy}\right)^2}\,dx \tag{7}$$

$$<\frac{\min}{X_0}\int_{X_0}^{X_0+\pi\sqrt{Ar_m}}\sqrt{1+\left(A\left(\frac{2\pi}{X_t}\right)\right)^2\cos^2\left(\left(\frac{2\pi}{X_t}\right)x\right)}$$

where X_0 is any value of x axis.
The number of sensors can be obtained dividing the length of the endoscope by the above L.

2-2 Realization theory of the profile of the endoscope

Profile must be calculated using the detecting curvature data ρ_j (j=1 $\cdot\cdot\cdot$ m). The flow chart of the realization theory of the profile of the endoscope is shown in Fig.3.

Curvature data

$$\rho_j=\frac{1}{r_j} \qquad (j=1\cdot\cdot\cdot m)$$

[1] Interpolation

k-th order polynomical approximation
for ρ_j (j= $\cdot\cdot\cdot$ m)
$$P(s)=a_ks^k+a_{k-1}s^{k-1}+\cdot\cdot\cdot$$
$$+a_1s+a_0$$

[2] Numerical Integration

$$\theta_{0i}=\int P_{(s)}ds \qquad (i=0\cdot\cdot\cdot n)$$
$$\overrightarrow{OB}=\overrightarrow{OA}+v_i$$
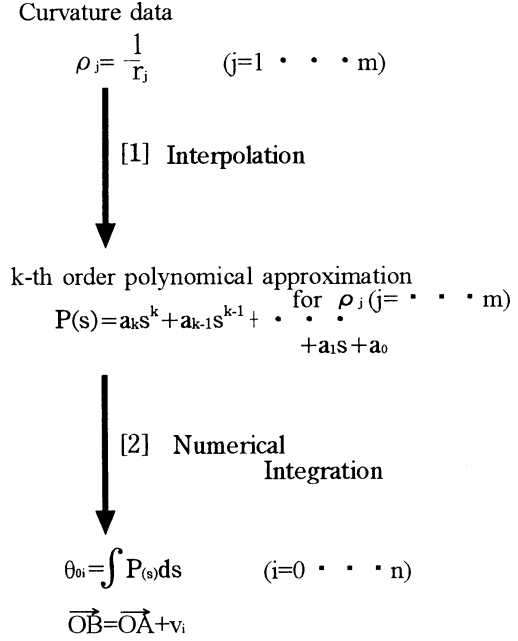
Fig.3 Calculation flow chart of the polynomial approximation of the curvature
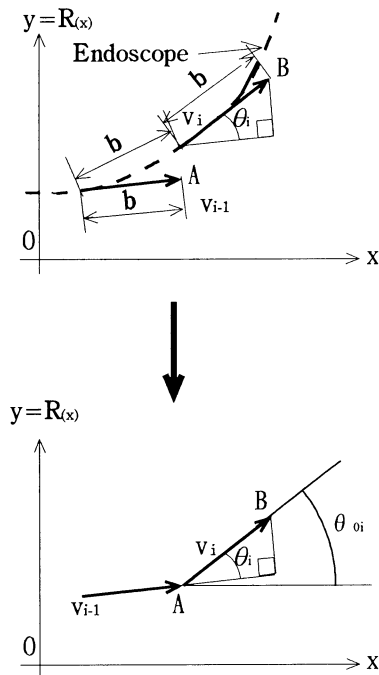
Fig.4 Directional vector for the calculation

Flow chart [1] shows that the polynomial function of curvature is calculated because a continuous function can be approximated by polynomial function according to Weierstrass's approximation theorem[2].

Flow chart [2] shows that the vector of the tangential slope of s axis on each point i with length b [m] is v_i, shown in Fig.4 where i is integer from 1 to n. $P(s_i)$ is the angle of the slope of each point as follows

$P(s_i) = \theta_i$

$$= a_k s_i^{k} + a_{k-1} s_i^{k-1} + a_{k-2} s_i^{k-2} + \cdots + a_1 s_i + a_0 \quad (8)$$

where θ_i is the angle on s axis. The total sum of the angle from 0 to i is as follows.

$$\theta_{0i} = \sum_{k=1}^{i} \theta_k \quad (9)$$

θ_{0i} is the angle from x axis at point i. So, the vector of v_i is expressed by the following equation.

$$v_i = (b\cos\theta_{0i}, \ b\sin\theta_{0i}) \quad (10)$$

If the vector of $\overrightarrow{OA}$ is defined as follow,

$$\overrightarrow{OA} = (x_{i-1}, \ y_{i-1}) \quad (11)$$

the vector of $\overrightarrow{OB}$ can be expressed as follows.

$$\overrightarrow{OB} = \overrightarrow{OA} + v_i$$

$$= (x_{i-1} + b\cos\theta_{0i}, \ y_{i-1} + b\sin\theta_{0i}) \quad (12)$$

3 Simulation

How bending profile could be calculated based on the realization theory was simulated. 2D endoscope was assumed as
 • Length   1.2 [m]
 • The number of sensors m=2, 3, 4, 5, 7, 9 and 11 points
 • The number of n=100, 500 and 1000
Reference profiles of endoscope are linear and circle of 2-dimension. Condition of simulation is shown in Table.1.

Table.1 Conditions of simulation

Figure	m	n
Fig.5 (a) Linear (b) Circle	11	100, 500, 1000
Fig.6 (a) Linear (b) Circle	2, 3, 4, 5, 7, 9, 11	1000

The simulation results are shown in Fig.5 and Fig.6.

The simulation results of n=100, 500 and 1000 (Fig.5 (a)) coincide with the reference profile of linear. The simulation results for n=100 and 500 (Fig.5 (b)) are different from the reference of circle however simulation results for n=1000 shows good agreement with the reference of circle. The simulation results of linear and circle (Fig.5 (a) and (b)) show the same profile of reference as the case of n=1000. The simulation results of linear of m=2, 3, 4, 5, 7, 9 and 11 (Fig.6 (a)) coincide with the reference profile of linear. The simulation results of circle (Fig.6 (b)) was same profile to simulation except the case of m=2.
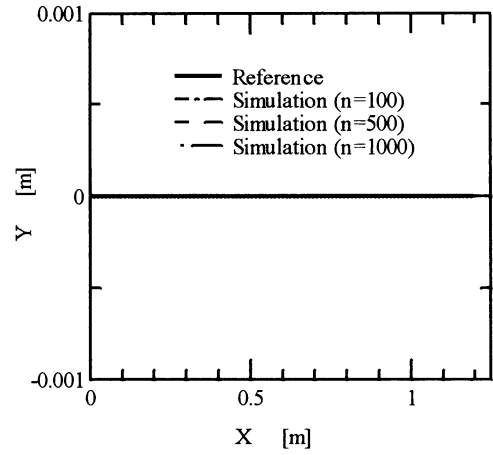
(a) Linear
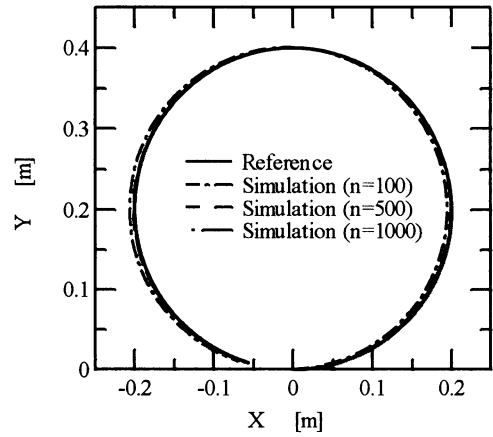
(b) Circle
Fig.5 Simulation Results (n=100, 500, 1000, m=11)
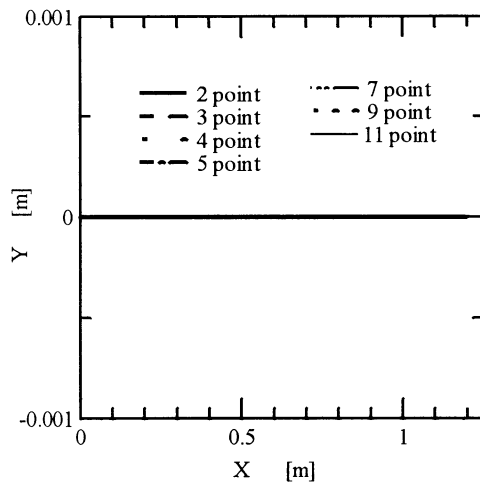
(a) Linear
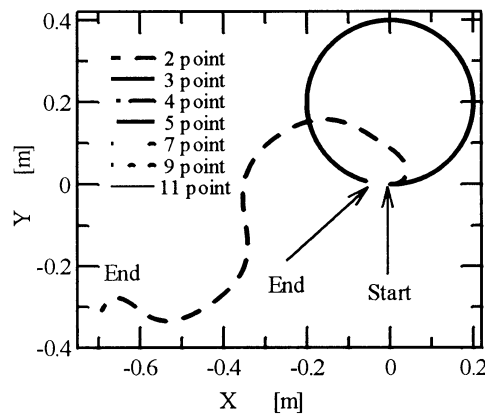
(b) Circle

Fig.6 Simulation Results (n=1000, m=2, 3, 4, 5, 7, 9, 11)

4 Experiment

Experiments of the realization of the profile of an endoscope using strain gauges were carried out.

4-1 Experimental method

The steel belt of length 1.2 [m], width 1.2 [cm] and thickness 0.15 [mm] was used instead of endoscope because to avoid the error due to bending in the direction of Z axis and moment. Strain gauges were pasted on the experimental 2D endoscope under the condition of L=120 [mm] and m=10.

Voltage from the strain sensors were amplified using differential amplifier. The signal from differential amplifier is inputted to a personal computer through A/D board.

4-2 Experimental evaluation

Experimental result of linear profile of 2D endoscope is shown in Fig.7. There is some deviation of the

experimental results from the reference because of the experimental error and calculation error. However the experimental results were shown almost same linear and circle profile.
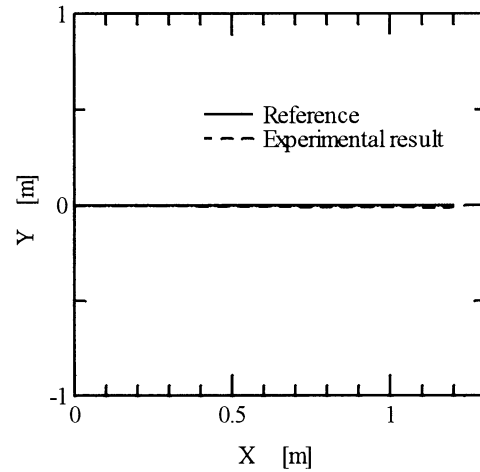
(a) Linear
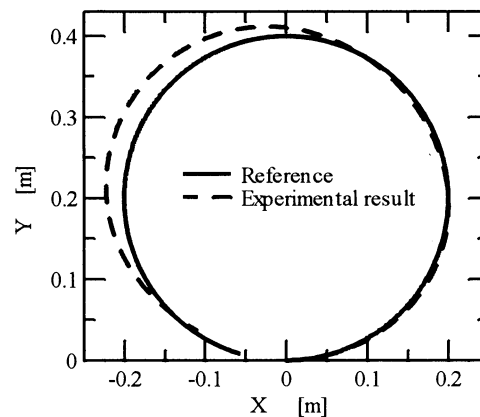
(b) Circle

Fig.7 Experimental Result (n=1000, m=10)

5 Conclusion

The simulation confirmed that bending profile of 2D endoscope could be calculated by the realization theory.

The experimental results shows the proposed system has high potential of realization of the profile of an endoscope for the navigation.

Reference

(1) Zyuri Yamashita, Masaaki Mochimaru, Yasushi Yamauchi, et all, "Evaluating A Real-Time Navigator System with Single Perspective View for Endoscopic Paranasal Sinus Surgery", Information processing experiment report Human Interface HI-74-5, pp.25-30, (1997), (in Japanese)

(2) Saburo Makinouchi, Tatsuo Torii, "Numerical Analysis", ohm Co., p.161, (1975), (in Japanese)

(3) Masami Ito, "Auto Control", Maruzen Co., pp.164-166, (1988), (in Japanese)

Development & Estimation of Dynamic Characteristic Model of Se nsor for Sensing Stiffness

T. J. Chung*, D. P. Hong**, H. Y. Kang**, S. Y. Hwang***, Y. S. Ku***

*Faculty of Mechanical Engineering Kunsan National University , Kunsan 573-701 , Korea

** Faculty of Mechanical Engineering, Chonbuk National University, Chonju, 561-756, Korea

***Dept of Mechanical Engineering Graduate School, Chonbuk National University, Chonju 561-756, Korea

Tel : +82-652-270-2374   Fax : +82-652-270-2460
E-mail: hongdp@moak.chonbuk.ac.kr / tjchung@kunsan.ac.kr / Kanghy@moak.chonbuk.ac.kr /
hwangsy@joy.hosanna.net / kuys@hosanna.net /

Abstract

This paper deals with sensor that has a sensing ability o f distinguishing stiffness. We have developed new signal p rocessing method that has distinguished different stiffness.

We estimated stiffness of object with new recognition in dex(R_{SAI}) that is ratio of SAI (Sensing Ability Index)

This paper describes our primary study for a new me thod of recognizing stiffness, which is need for precisio n work system. This is a study of dynamic characteristi cs of sensor, new method (R_{SAI}) have the sensing ability of distinguishing materials. Experiment and analysis are executed for proper sensing condition. First, we develo ped advanced sensor. Second, we developed new metho ds that have a sensing ability of distinguishing stiffness. Dynamic characteristics of sensor are evaluated through new recognition index (R_{SAI}) that is ratio of sensing ab ility index.

Distinguishing of object is executed with R_{SAI} method relatively. We can use the R_{SAI} for finding stiffness of object. Applications of this method are finding abnormal condition of object (auto-manufacturing), feeling of obj ect (medical product), robotics, safety diagnosis of struct ure, etc.

1. Introduction

So modern industry society is developed, Auto manuf acturing is a general tendency. Need of new mechatroni cs become growing for work in dangerous area, pollutio n area, high speed, and high precision. Interests in deve lopment of sensor that acquires information for object b ecome concentrated for precision work. Accordingly, de velopment of sensor that acquires the information for ob ject by simple and easy improvement of productive cap acity is expected. A study of the characteristics of a sof t sensor that has the sense of human's skin. And then a nalysis of experiment is only used out-voltage value[1]. In another study, it is analyzed to obtain the maximum signal strength, the total number of concaves and conve xes of signal, and the number of crossings with the line of mean signal strength. Each set of the parameters obt ained is compared with the training set of data in the n eural network, and the material of the measured sample is fixed[2]. A study is concerned with the development of a palpation probe for measuring body softness such as the prostate cancer and hypertrophy. In order analysis, the discrete wavelet decomposition method was introdu ced to deal with the signal[3]. A study is dynamic char acteristics of sensor, which have the sensing of a materi al. And then we developed fundamental sensor and new method that makes an estimate of enhanced sensing ab

ility. Dynamic characteristics of sensor is estimated, eval uated through new method (SAI) that use the power spe ctrum density[4].

Above descriptions, analysis methods of many studies are used with various methods. But analysis methods t hat are compared with experiment results are analyzed with FEM or neural network, wavelet method and so on. These methods are not sufficient to distinguish stiffness of object. Ability of recognition is not sufficient too. A nd then, we want to develop new method that is used f or design of sensor, and optimization of sensing ability. We developed the new recognition index (R_{SAI}) that is r atio of SAI (sensing ability index) for recognizing stiffn ess of object. The new recognition index (R_{SAI}) has abil ity to distinguish stiffness of object exactly. Also the re cognition index (R_{SAI}) is real time analysis system. Disti nctive characteristics of new recognition index (R_{SAI}) is energy concept that analyzes the result of experiment wi th power spectrum density. The method has ability of di stinguish for object and is capable of evaluating dynami c characteristics of sensor. We will use the R_{SAI} for find ing stiffness of object. Applications of this method are f inding abnormal condition of object (auto-manufacturin g), feeling of object (medical product), robotics, safety diagnosis of structure, etc.

2. Experimental Equipment & Experimental Method

2.1 Development of Sensor

Hard-Hard type (HH) sensor is made for experiment. Sp ecification of sensor is displayed below (Table1). Schemati c of sensor is show in Fig 1. Array of PVDF film on the first floor is show in Fig 2. Array of PVDF film on the second floor is show in Fig 3.

Table 1. Specification of HH Sensor

Base 1 material	Base 2 material	Surface m aterial (thickness)	Sensor Type
Hard	Hard	0.5 mm	HH
PVDF film symbol			
PVDF1, PVDF2			HH1
PVDF3, PVDF4			HH2
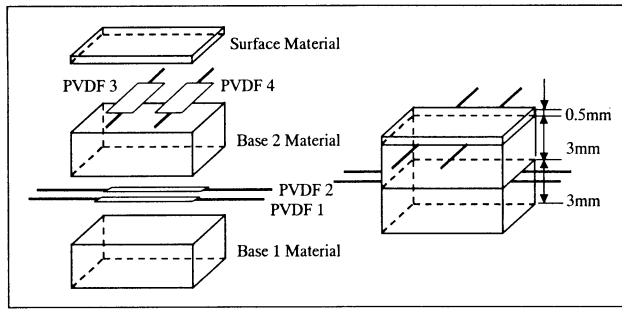
Fig 1. Schematic of Sensor
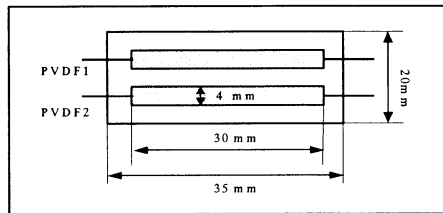
Fig 2. Schematic of Array of PVDF Film on the First Floor
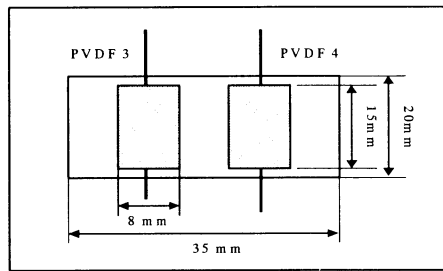
Fig 3. Schematic of Array of PVDF Film on the Second Floor

2.2 Experimental Equipment System

A schematic of experimental equipment system is shown i n Fig 4. Input controller that changes the value of displace ment, acceleration (g) and frequency control the Input syst em. Input system's function is execution that input conditi on is indicated by input controller. Sensor is attached by S ensing attachment system. Sensing data is transferred to si gnal processing & analyzing system by data transfer modu le. Sensing data is analyzed and evaluated by signal proce ssing & analyzing system.
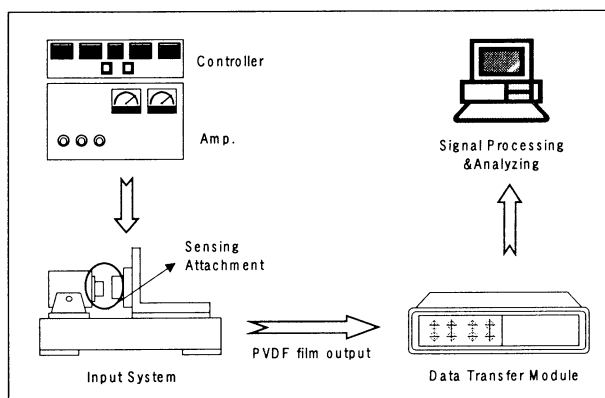
Fig 4. Schematic of the experimental set- up

We prepared four types of object for experiment. Four objects

have the different stiffness. There are Steel (Fe), Aluminum (Al), Wood (Wo), and Sponge (Sp). The diameter of object is 10mm. The thickness of object is 3 mm. Schematic of object is show in Fig.5
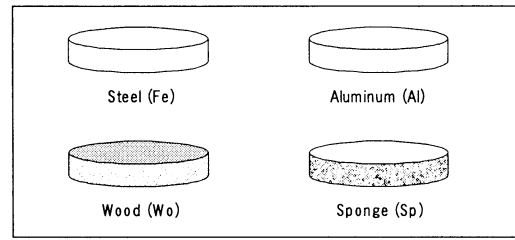
Fig. 5 Schematic of Object

2.3 Method of Experiment and Analysis

Experiments are executed as follow:
A) Object Recognition analysis according to acceleration (g) changing using the new R_{SAI} method
B) Object Recognition analysis according to frequency changing using the new R_{SAI} method
C) Object Recognition analysis according to displacement changing using the new R_{SAI} method

A schematic of analysis procedures is shown in Fig 6.
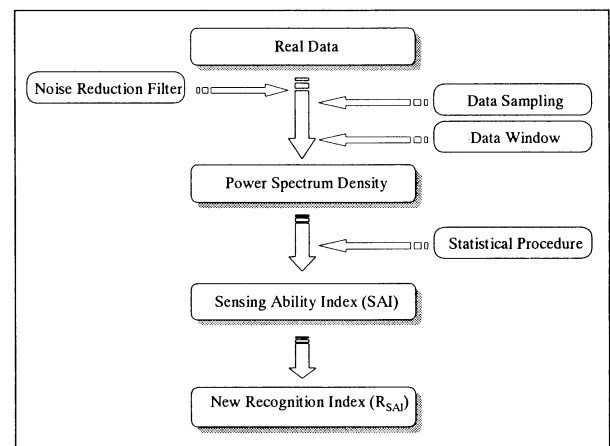
Fig 6. Schematic of Analysis Procedures

Changing input variables are show in Table 2.

Table 2. Input Variables for Experiment

Changing variable	Changing value
Acceleration	1g, 5g, 10g
Frequency	10Hz, 20Hz, 30Hz
Displacement	0.1:0.2:1.3

3.Result and Discussion

3.1 Object recognition analysis according to acceleration (g) changing using the new R_{SAI} method

a) In 10Hz – 0.7 mm (Fig. 7-a), according to (g) changing value of R_{SAI} is almost not changing. And then the recognition index (R_{SAI}) is not concerned with acceleration (g). About ability of distinguishing for

object, 1g is higher than others. 10g has different character that R_{SAI} of wood is greater than sponge. It is reason that the sensor has the critical point.

b) In 20Hz – 0.7 mm (Fig. 7-b), according to (g) changing value of R_{SAI} is almost not changing, too. And then the recognition index (R_{SAI}) is not concerned with acceleration (g). About ability of distinguish for object with R_{SAI}, 1g is better as compared with others(5g, 10g). It is reason that too much input force is not good for sensing condition.

c) In 30Hz – 0.7 mm (Fig. 7-c), according to (g) changing value of R_{SAI} is almost not changing, too. And then the recognition index (R_{SAI}) is not concerned with acceleration (g). In this case, every have a good ability of distinguishing for object. Also, 1g is better as compared with others (5g, 10g).

d) Value of R_{SAI} is $Sp > Wo > Al > Fe$, we can distinguish stiffness of object among them using R_{SAI} method. (Fig 7.a-c)

e) In this case, a tendency of R_{SAI} value is similar to others. And then the recognition index (R_{SAI}) are not concerned with acceleration (g). (Fig 7.a-c)

f) Comparing R_{SAI} value of object, we can find that 1g has better sensing ability than other conditions.
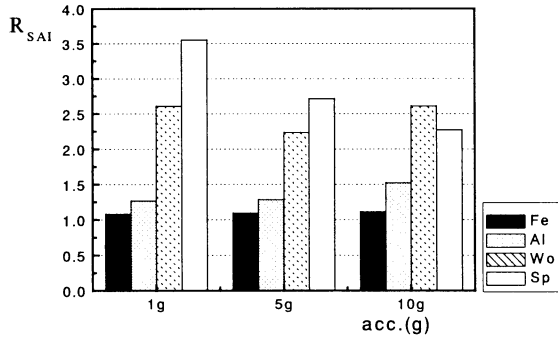
Fig.7-a The R_{SAI} Value According to Acceleration (g) Changing (10 Hz – 0.7mm)
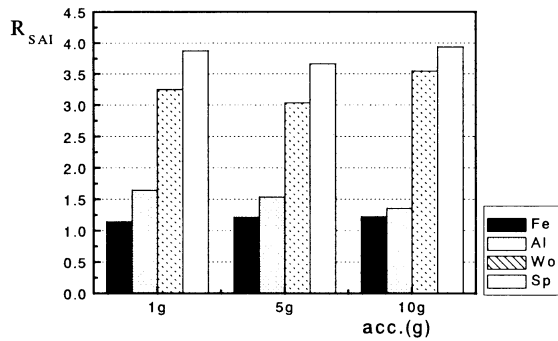
Fig.7-b The R_{SAI} Value According to Acceleration (g) Changing (20 Hz – 0.7mm)
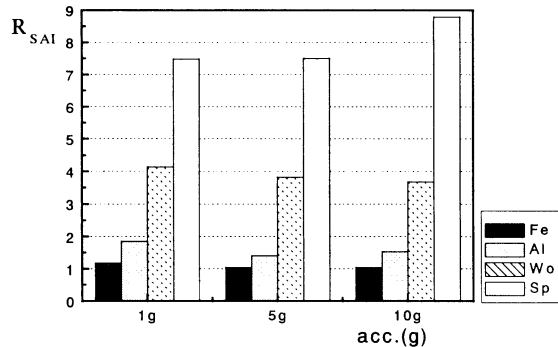
Fig.7-c The R_{SAI} Value According to Acceleration (g)

Changing (30 Hz – 0.7mm)

3.2 Object recognition analysis according to frequency changing using the new R_{SAI} method

a) In 1g – 0.7 mm (Fig. 8-a), according to frequency changing tendency of R_{SAI} is almost not changing. But ability of distinguishing for object, 30Hz is higher than others. And then the recognition index (R_{SAI}) is concerned with frequency. It is reason that the sensor is very sensitive to frequency.

b) In 1g – 0.9 mm (Fig. 8-b), according to frequency changing tendency of R_{SAI} almost not changing, too. But ability of distinguish for object, 30Hz is higher than others, too. And then the recognition index (R_{SAI}) are concerned with frequency. In recognition of object, proper sensing condition is very important. In this case, frequency is very important to distinguish stiffness of object.

c) Value of R_{SAI} is $Sp > Wo > Al > Fe$, we can distinguish stiffness of object among them using R_{SAI} method. (Fig 8.a-b)

d) In Fig. 8, we can find that the new R_{SAI} method will use for finding optimal point of recognition ability.

e) Comparing R_{SAI} value of object, we can find that 30Hz has better sensing ability than others conditions.

f) We can find that the new R_{SAI} method evaluate stiffness of object among them according to stiffness changing.
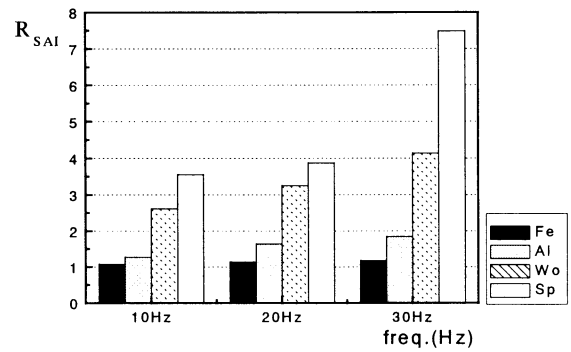
Fig.8-a The R_{SAI} Value According to Frequency Changing (1g – 0.7mm)
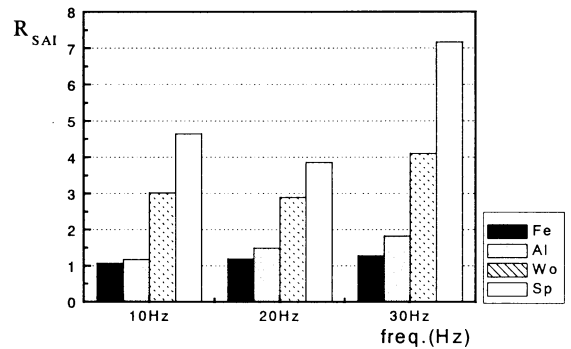
Fig.8-b The R_{SAI} Value According to Frequency Changing (1g – 0.9mm)

3.3 Object recognition analysis according to displacement changing using the new R_{SAI} method

a) In 10Hz -1g (Fig. 9-a), according to displacement changing tendency of R_{SAI} is almost not changing. And then the recognition index (R_{SAI}) is not concerned with displacement. About ability of distinguishing for object, (Fig.9-c) 0.5mm is lower than others. 0.5mm has different character that R_{SAI} of wood is greater than

sponge. It is reason that the sensor has the critical point.

b) In 20Hz – 1g (Fig. 9-b), according to displacement changing value of R_{SAI} is almost not changing, too. And then the recognition index (R_{SAI}) is not concerned with displacement. About ability of distinguish for object with R_{SAI}, 0.7and 0.9mm are better as compared with others. It is reason that too much input displacement is not good for sensing condition

c) In 30Hz – 1g (Fig. 9-c), according to displacement changing value of R_{SAI} almost not changing, too. And then the recognition index (R_{SAI}) are not concerned with displacement. In this case, every have a good ability of distinguish for object. Also, 0.7and 0.9mm are better as compared with others.

d) Value of R_{SAI} is $Sp > Wo > Al > Fe$, we can distinguish stiffness of object among them using R_{SAI}

e) In this case, a tendency of R_{SAI} value is similar to others. And then the recognition index (R_{SAI}) are not concerned with displacement. (Fig 9.a-c)

f) In recognition of object, proper sensing condition is very important. In this case, displacement is very important to distinguish stiffness of object.

g) In Fig. 9, we can find that the new R_{SAI} method will use for finding optimal point of recognition ability.

h) Comparing R_{SAI} value of object, we can find that 0.7and 0.9mm have better sensing ability than others conditions.

i) We can find that the new R_{SAI} method evaluate stiffness of object among them according to stiffness changing

4. Conclusion

We have made the sensor (Hard –Hard type) for di stinguishing stiffness of object. And then we proposed n ew analysis method (R_{SAI}) that uses the power spectrum density technique for distinguishing stiffness of object. The new recognition index (R_{SAI} method) has a good ab ility of distinguishing among different object. R_{SAI} meth od has a good ability of distinguishing for response qua lity among different objects. Using the characters of R_{SA} $_I$ method, we can design the best optimal sensor having a good sensing ability for distinguishing stiffness of ob ject. The new recognition index (R_{SAI} method) has the b est ability of distinguishing in range of 30Hz – 1g –0. 7mm and 30Hz – 1g –0.9mm. Applications of R_{SAI} met hod are various field. We can use the R_{SAI} for finding s tiffness of object. Applications of this method are findin g abnormal condition of object (auto-manufacturing), fee ling of object (medical product), robotics, safety diagnos is of structure, etc.

References

[1]Seiji Chonan, Zhong-Wei Jiang, Jun Ibayashi and Mesa yuki Sato, "Development of Soft Tactile Sensor (1st Repor t, Fundamental Structure and Characteristics of the Senso r)" Transactions of the Japan Society of Mechanical Engin eers, vol. 60. no. 580., pp. 4203-4210, 1994.
[2]Seiji Chonan et al, "Development of Tribosensor system Using Neural Networks (Tribosensor Using Piezoelectric Ceramics)" Transactions of the Japan Society of Mechanic al Engineers, vol. 61. no. 587C., pp. 354-361, 1994. 1995
[3] Zhongwei Jiang, Seiji Chonan, Yoshikatu Tanahasi, "St udy on Development of Prostatic Palpation Probe and Mea surement of Body Softness",Proceeding of the 3rd IWAM, December 1999., pp218-223,1999
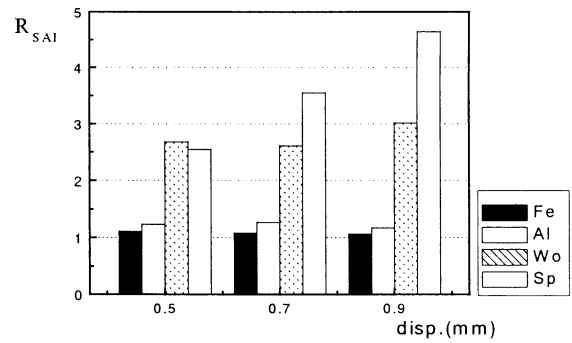
Fig.9-a The R_{SAI} Value According to Displacement Changing (10Hz -1g)
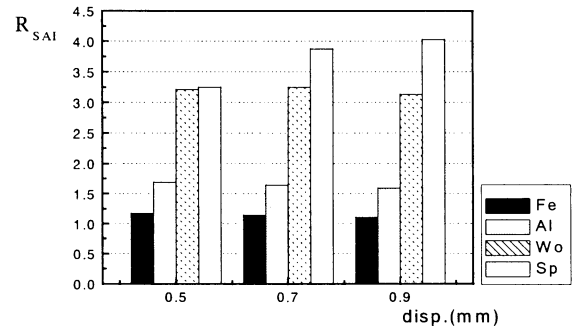
Fig.9-b The R_{SAI} Value According to Displacement Changing (20Hz -1g)
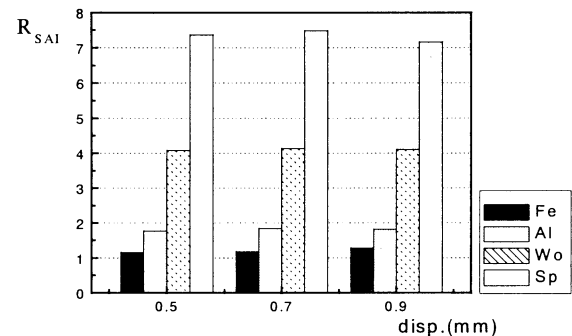
Fig.9-c The R_{SAI} Value According to Displacement Changing (30Hz -1g)

[4]Dong Pyo Hong, Hee Yong Kang, Seong Youn Hwang, et al. " Dynamic Characteristics Analysis of Sensor for E nhance Sensing Ability ", Proceeding of the 3rd IWAM, D ecember 1999., pp 366-371,1999
[5]International Organization for Standardization, Evaluatio n of Human exposure to whole-body vibration - part1: Ge neral Requirements (ISO 2631/1), International Organizatio n for Standardization, 1985.
[6]Association of German Engineers, Effect of Mechanical vibrations on Human beings (VDI 2057), Association of G erman Engineers, 1987.
[7]British Standards Institution, British Standard Guid e to Measurement and Evaluation of Human Expose to Whole-body Mechanical Vibration and Repeated S hock (BS 6841), British Standards Institution, 1987.
[12]M. J. Griffin, Handbook of Human Vibration, Academi c Press Inc., 1990.

Characteristics of a Planar Type Micro Electro Magnetic Sensor for Inspection of Metal Cracks

Xianhe DING, Takao HASHIDA and Katsutoshi KURIBAYASHI
Faculty of Engineering,Yamaguchi University
2-16-1 Tokiwadai UBE,755-8611 Japan

ABSTRACT

In this paper, micro electro magnetic sensors have been developed for inspection of metal cracks. With help of micromaching techniques, a 3000-μ m-diameter microcoil has been fabricated. Comparison with past sensors developed, the sensor has characteristics of minimum size, spiral planar type and ultrathick structure because of using high aspect ratio photoresist SU-8 and Ni electroplating method. With testing circuit, the static characteristics of the microsensor and cracks of a metal plate were measured. The experimental results show the micro electro magnetic sensor can effectively detect the cracks on the metal plate.

Keywords: microcoil, eddy current, electro magnetic sensor, small diameter pipe, crack, high aspect ratio

1 INTRODUCTION

To inspect the inner cracks of metal pipes under 1 inch diameter, micro electro magnetic sensors have been developed. In the past, sensors for inspection of metal cracks were developed such as positional sensors, supersonic sensors and laser sensors, but the sensors can not satisfy the practice detecting needs to the micropipes, because their size is too large [1][2].

In this paper, as the micro electro magnetic sensor, a planar type microcoil of the outer diameter 3000 μ m has been fabricated successfully. By using the high aspect ratio photoresist SU-8 [3] and Ni electroplating method, the microcoil with ultrathickness 300 μ m has been achieved in the fabricating process, thus detecting sensitivity of the microsensor for cracks of micropipes has been improved. The static electric characteristics and output functions as the crack width of 0-5mm have been measured.

2 PRINCIPLE and FABRICATION

2.1 Principle of the Sensor
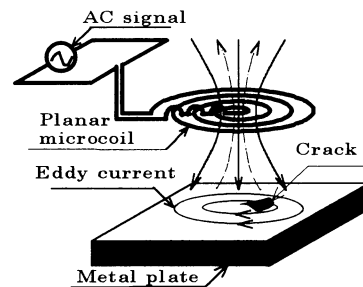
Fig1 Principle of the planar type

micro electro magnetic distance sensor

The micro electro magnetic sensor is based on the measurment of the microcoil inductance variation. Fig1 shows its principle diagram.

When the microcoil is supplied high frequency AC signal, it will generate electromagnetic field which induces an eddy current on the metal plate, the eddy current will lead to a variation of microcoil inductance. Since cracks on the metal plate affect the eddy current value, their presence can be detected from the variation of the coil inductance.

If the metal plate is regarded as a short circuit coil, we can get its equivalent circuit in [Fig 2].
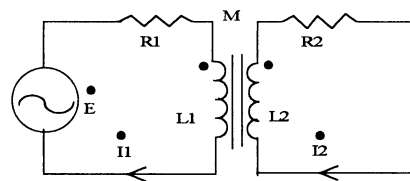
Fig2 Equivalent circuit of the Fig1

$$R1*\dot{I1}+j\omega*L1*\dot{I1}-j\omega*M*\dot{I2}=E \qquad (2.1)$$

$$-j\omega*M*\dot{I1}+R2*\dot{I2}+j\omega*L2*\dot{I2}=0 \qquad (2.2)$$

Thus equivalent impedance of the microcoil

$$|\dot{Z}| = |\dot{E}/\dot{I1}| =R1+R2*Y+j(\omega*L1-\omega*L2*Y) \qquad (2.3)$$

$$\{Y=\omega*\omega*M*M/(R2*R2+\omega*\omega*L2*L2)\}$$

Equivalent inductance of the mcrocoil

$$L = L1-L2*Y \qquad (2.4)$$

R1-----Coil resistance   R2-----Metal resistance

E------AC signal   L1-----Coil inductance

L2-----Metal inductance   M------Mutual inductance

I1-----Coil current   I2-----Metal current

ω-----Frequency

2.2 Fabrication of the Microcoil

2.2.1 Photomask Design

In the fabrication process of the microcoil, a photomask design is necessity, because the size and shape of the microsensor is decided by photomask .

Generally the microcoil is requested to have a high inductace and low resistance. In comparison to different planar coil structures[4], the spiral coil can obtain the largest inductance, and large inductance also depends on increasing the total number of the microcoil turns. The low resistance can be obtained by increasing the width and thickness of the coil[5].

Considering high inductance, low resistance and small size, the final size and shape of microcoil is presented in Fig3 and Table1.
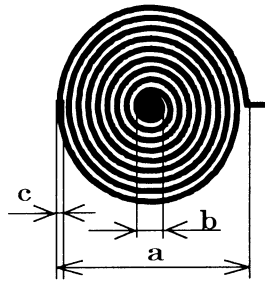
Fig3 Mask Sample

Table 1 Microcoil size

a(Maximum diameter of a coil)	3000(μ m)
b(Center diameter of a coil)	300(μ m)
c(Width)	100(μ m)
Thickness	300(μ m)

2.2.2 Fabrication Process of the Microcoil

A new fabrication process of the microcoil is schematically described in Fig4.

1) The seed layer (Cu/1 μ m) were sputtered on the glass plate.

2) Photoresist SU-8 is deposited on the Cu wafer and by photomask, the photoresist is exposed under UV light.

3) Development is done in PGMEA.

4) The high aspect ratio microstructure of microcoil is

achieved by Ni electroplating using ultrathick photoresist SU-8 mold.

5)The photoresist SU-8 is removed.

6)Cu is etched.

The microcoil sample fabricated is shown in Fig 5. It has high aspect ratio and ultrathick structure.
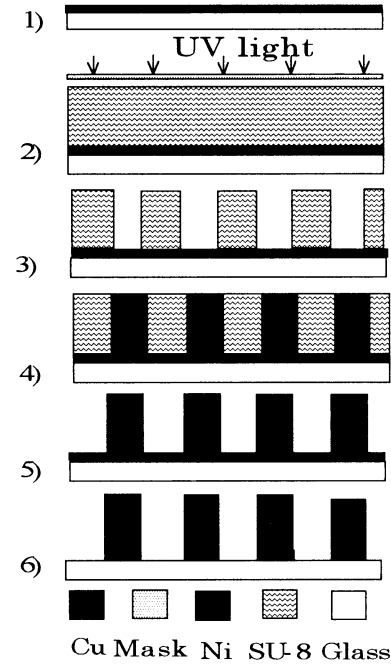
Fig4 Fabrication process of the sensor
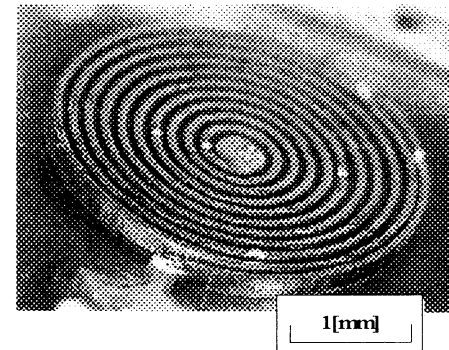
Fig5 Microcoil sample fabricated

3 THE STATIC CHRACTERISTCS

3.1 The Measurement of the impedance characteristics of the sensor.

In the Fig6, by controlling the position of XYZ stage along Z direction, the variation of the microcoil impedance can be measured from the impedance meter.

W---crack width   Y---moving direction coordinate

H---hight between coil and metal plate

When the V is only changed from 1mm/s to 10mm/s, the output of the sensor has same result. That means a change of the V under 10mm/s doesn't influence the crack inspection.

When the H, θ ,or W is only changed, the outputs of the sensor are shown in Fig11, Fig12 and Fig13. By the experimental results, following conclusions can be obtained.

1)The more near metal suface is, the more high detecting sensitivity has.

2)As following θ increasing, slope of the output curve is reduced.

3)For different W, the output of the sensor has obvously changes.
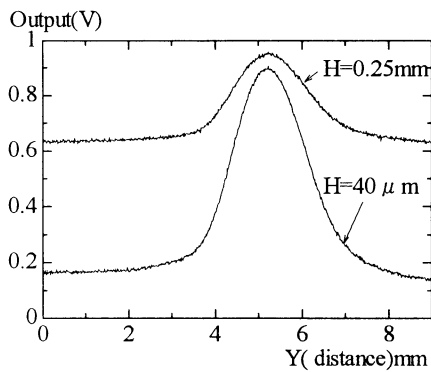
Fig11 Y as a fuction of the sensor output at V=1mm/s,

θ =0, W=2mm
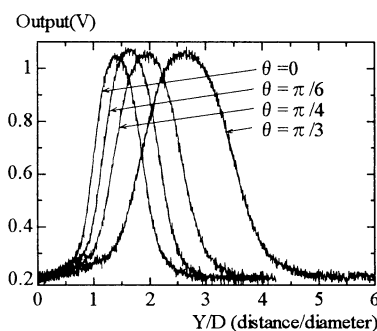
Fig12Y/D as a fuction of the sensor output at H=40μ m,

V=1mm/s,W=3mm.
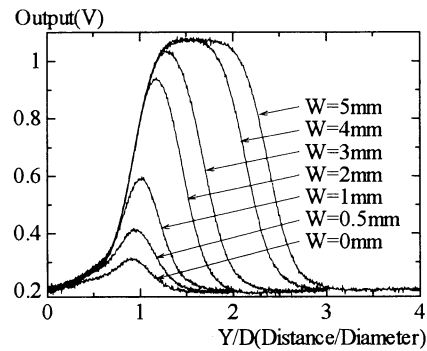
Fig13 Y/D as a fuction of the sensor output at H=40μ m,

V=1mm/s, θ =0

5CONCLUSIONS

1) Micro electro magnetic sensor using microcoil was developed for inspection cracks of the metal plate.

2) A microcoil of thickness of Ni 300μ m, and outer diameter 3000μ m was fabricated successfully by using ultrathick high-aaspect-ratio photoresist SU-8 and Ni electroplating method.

3) The static and crack detection characteristics of microcoils have been measured by detecting circuit. Experimental results show the microcoil can detect the cracks of the metal plate.

REFERENCES

[1]Y.Hamasaki and T.Ide (1995) "a Multi-layer Eddy Current Micro Sensor for Non-destructive Inspection of Small Diameter Pipes" Trnsducers'95, pp.136-139

[2]Y.Hamasaki and T.Ide (1995) "Fabrication of Multi-layer Eddy Current Micro Sensors for Non-destructive Inspection of Small Diameter Pipes" IEEE pp. 232-237

[3]G.Engelmann, O.Ehrmann, J.Simon and H.Reichl (1992) "Fabrication of High Depth-to-Width Aspect Ratio Microstructures" MEMS' 92, pp. 93-98

[4]K.Kawabe, H.Koyama and K.Shirae (1984) "Planar Inductor" IEEE Trans. Magn, MAG-20.5, pp1804-1808

[5]K.Muraki and K.Kuribayashi (1998) "New Fabrication Method of Multiple-Layer Microcoil Using Anodic Oxidized Aluminum" PROC. Advanced Mechatronics, pp.484-488

Fig7 shows the microcoil impedance measured as a function of the distance between the microcoil and metal plates which are stainless, copper and aluminium ones. Comparision with copper and aluminium, stainless has better static characteristics, because its impedance has largest difference between H=0and H=3mm
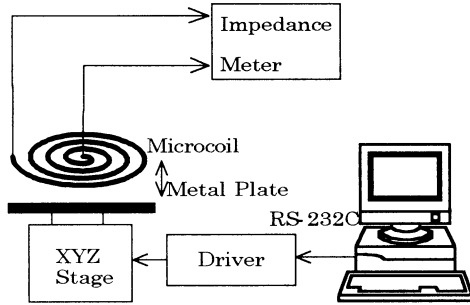
Fig6 Testing system frame of the impedance characteristics
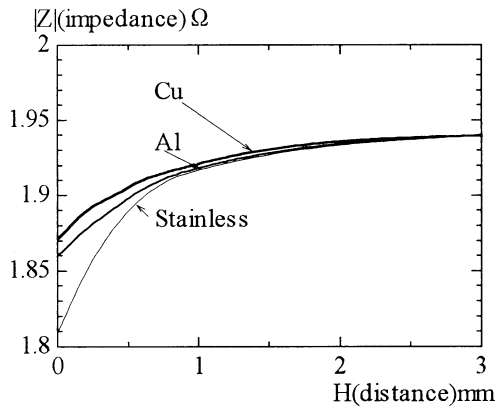
Fig7 Z-H Experimental result of the
static characteristics of the sensor

3.2 The Measurement of output characteristics of the sensor
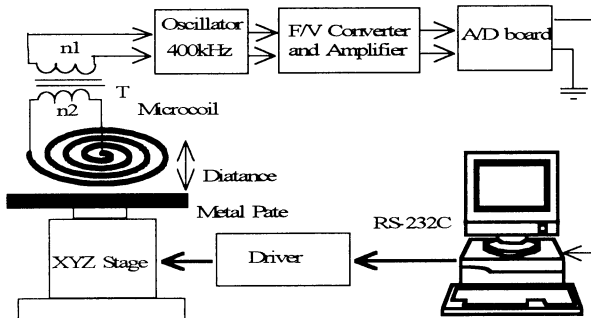
Fig8 Experimental system to measure output
voltage characteristics of the sensor

The Fig8 shows the testing system of the output chatacteristics of the mirocoil. To achieve high sensitivity, the tranceformer T is used, $n1/n2=80/1$

Firstly the computer controls the position of XYZ stage, at the same time, by F/V and A/D transformation, the microcoil output will be obtained. The curve of distance with the voltage of the microcoil is shown in Fig9. Fig9 shows that the microcoil has better output characteristcs at distance of 0—1mm and among three metal plates it has high sensitivity to stainless.
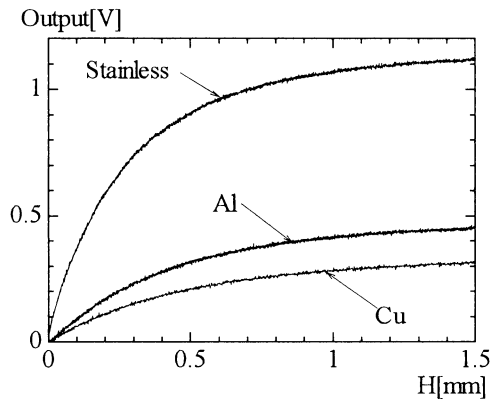
Fig9 Stainless ,Al and Cu output as a function of distance

4 CRACKS DETECTION OF THE METAL PLATES
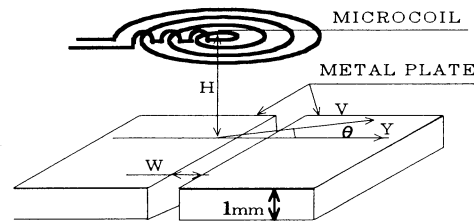
Fig10 The structure diagram for detection
cracks of metal plates

In order to obtain the metal crack detecting chracteristics of the microcoil, the following experiments were done. Its structure diagram is shown in Fig10, and stainless is as experimental detecting material. Experimental detecting circuit is the same as Fig 8. V---moving speed θ ---direction angle

Human Motion Recognition in Intelligent Space

Takahiro Yamaguchi, Joo-Ho Lee and Hideki Hashimoto
Institute Industrial Science, University of Tokyo
7-22-1-Roppongi Minato-ku, Tokyo 106-8558, Japan
E-mail: {guch, leejooho, hideki}@vss.iis.u-tokyo.ac.jp

Abstract

The Intelligent Space is an environmental platform that enables to support human life. The Intelligent Space can offer many useful support for human, such as baggage carrying, guidance of the way, security guard of the room and etc. Since in order to support human, we need to know what is happening in the room. Sensing devices are important elements in the Intelligent Space. We develop a human motion recognition system using vision-based sensors with some artificial inference.

We describe the stereo vision system that is capable of tracking the 3D movements of several human in real time and give experimental results obtained in a real-world environment.

1. Introduction

1.1 Intelligent Space

Intelligent Space is an area such as a room, a corridor or a street that is equipped with sensors and actuators [1]. The sensors have the main purpose to perceive what is happening in the space, especially what humans are doing. They include active and passive cameras, microphones, microphone arrays and tactile sensors. Actuators are currently mainly used to provide information to the inhabitants. This is done with speakers, screens, pointing devices, switches or robots inside the room [Fig.1].

1.2 3D Human Motion Tracking

To detect what is happening inside the room, the Intelligent Space tracks the movements of human. For this purpose, a vision sensor is the best sensor. Recognizing the human is done in two steps. First the area or shape of a human is separated from the background. Second features of the human such as head, hands, feet, eyes etc are recognized with their colors. From the images, taken by several cameras, we estimate the 3D position of human and extract meaningful data from the row image data[2, 3].

1.3 Related Work

There are many concepts of distributing intelligence into a space and supporting human. In such kind space,

sensing devices are important elements and among the several sensors, vision sensor is the most useful device. Many methods of using image sequence such as optical flow [4], extracting silhouettes [5] and stereo vision tracking [6] have been proposed. Since optical flow calculation need so much PC power, it becomes hardware problems. Extracting silhouettes is useful for estimating human gesture, however it becomes template matching problems. Stereo vision tracking is simple and need less calculation. We have constructed 3D tracking vision system. We extract what human is doing from raw data with simple calculation.

2. System Overview

2.1 System

To detect human motion, we adopt vision system [6]. Camera servers perform skin color detection [Fig.2]
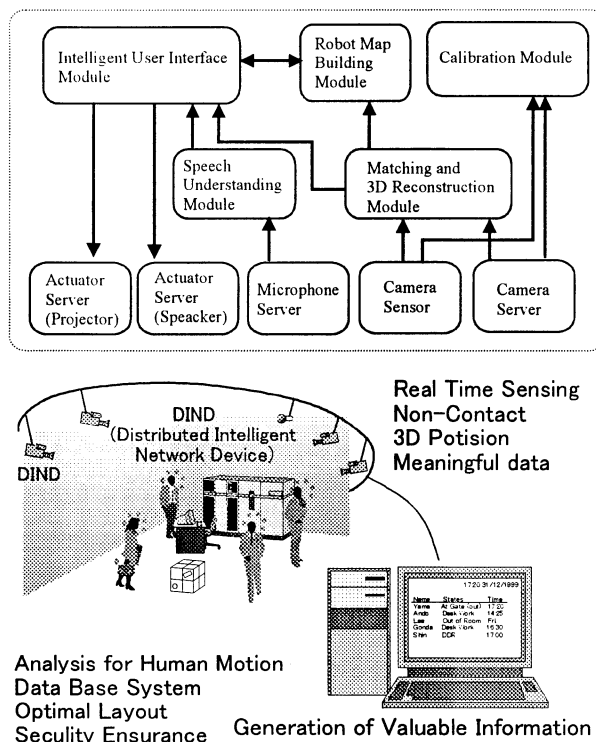
Fig.1   Intelligent Space

and background separation. They plot human positions on 2D frames. Matching and reconstruction module matches the clusters which camera servers made, and reconstructs 3D Human part position. Fig.3 shows the real-time plotting window by 2D (x, y axis). Human models such as human physical data and his motion characteristics are applied and the system generates valuable data.

Background separation Moving object, watched by a static camera, in front of a static background viewed with a static camera can be identified using background separation.

Skin color detection By segmenting the shape into colored regions or looking for skin colored regions, hands and head are identified directly.

Performance
3D position error: under 5 [cm]
Covering range of the system: 4[m]×3[m]
    (It depends on the orientation of cameras)
Problems:
    Manual calibration
    Varying of light source
Characteristics:
    Robust to real world as shown in Fig.2

System Architecture
Machine: Pentium II 300MHz
          And Pentium MMX 200MHz
OS: Linux 2.0.35
CCD Camera: VC-580(Chinon)
Programming Languages: C++ and Tcl/Tk
Soft Architecture: shown in Fig.4

2.2 Applications
2.2.1 the Rigid Security Situation
Rigid Security Situation can be considered as an application such as in banks, night buildings, etc.

Fig.2 Example of Blobs Extraction
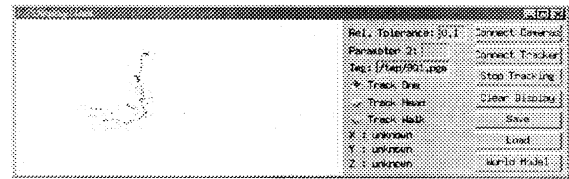
Fig.3 Real-time Display Window
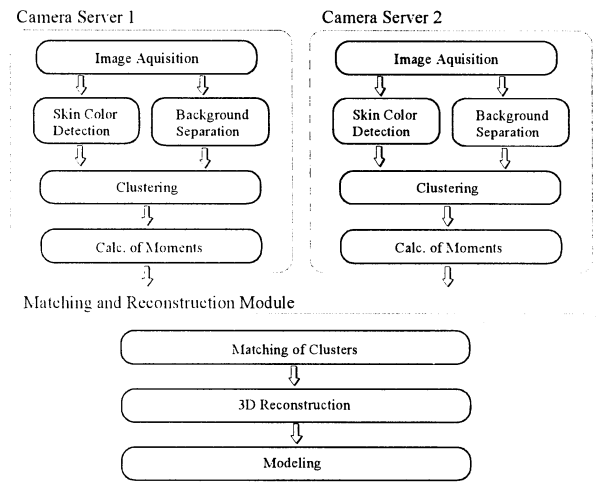
Fig.4 Flow of Data

Vision system recognizes individuals such as guards and others. It is useful for security.
2.2.2 Prevention of Crimes
This system is applied in prevention of crimes in the place where many people come such as grocery stores and department stores. There are many shoplifting in such kind stores. Since our proposed system stores some visual data and human motion data, and constructs databases of each people's motion trajectory, the strange actions can be detected [2].
2.2.3 the Optimal Layout of facilities at office and shopping booth
Functional layout influences human comfort in office. This system detects human position, stores the data, analyzes room usability and provides useful data for layout. For example, there is a point where people are always crowded, and if the point is more wider people can walk through comfortably. Moreover, in the case of shops, the system observes customers attitudes and analyzes their interests automatically, it is possible to change the layout of the space.

2.3 Required functions of 3D tracking system
In order to realize applications that is proposed in section 2.2, we need some functions in 3D tracking system to detect individuals, to watch where he is and

to store individuals motion, action and gesture data. If we get the information where he is interested in most, will generate valuable data.

3. Database System and Replay System

The system has the databases and reserves some data such as individuals motion information and some other data. The database includes time records and state of individuals, so that the system reproduces what was happening in the room [Fig. 5]. First we use simple columnar shape model. (In section4 we detect individual height and estimate who the man is.) The system compresses the normal visual data into state information. If the system stores visual data, the system needs huge capacity. Moreover the proposed system reserves a few main visual data at the entrance of the room and almost of visual data are abandoned. Since the system extracts necessary information, it's enough to reproduce what was happening in the room at any time.
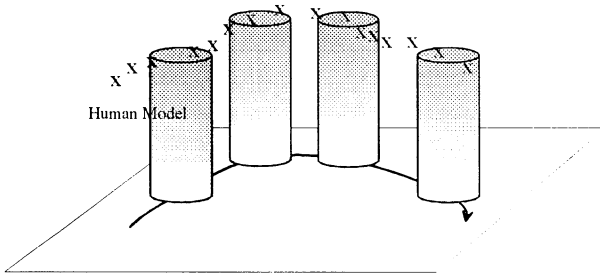
Fig.5   Human Shape Model and Reconstruction

4.   Human Comprehension

4.1   Head Position

Head Position is important in estimating human position, because it means human position. The system mainly detects active objects. To estimate head position, there are two methods as follows:

1)   Separation head data from others
2)   Smoothing the output

There are some blobs not only face, but also hands, feet, and so on. We have to separate faces from others [Fig.6 (a),(b)]. System outputs are noisy, because of sensor and human characteristics. As human walks, light condition varies. The system detects some blobs on the face and the output of the system may include noise. The output including noise is shown in (c). We apply a smoothing calculation and generate (d). We could understand human motions directly with

(a) output without   (b) extraction of head
     extraction

(c)   output xy plot   (d) smoothing data
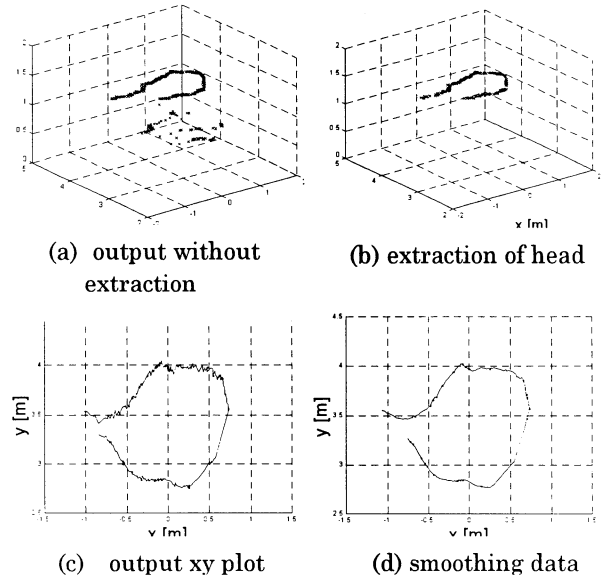
Fig.6 Head extraction and smoothing

smoothing data. However   an appliance of smoothing is sometimes not good in analyzing human by the systems, since smoothing abandons some human characteristics included in row data. The noise shown in Fig.6(c) is appeared in detecting a walking human. Row data includes meaningful information such as human is walking or moving his hands and etc in. They will be analyzed as a future work of this project.

4.2   Velocity of Walking Human

In the estimation of head position, we only think about position. If the system can get velocity of walkers, it is a good clue for assumption of their state such as human hasty, busyness, and etc. We usually walk about 4 [km/hour] on roads, but inside buildings we can't walk so fast. In the room we walk less than 1.0 [m] in a second, and in order to detect position changes we
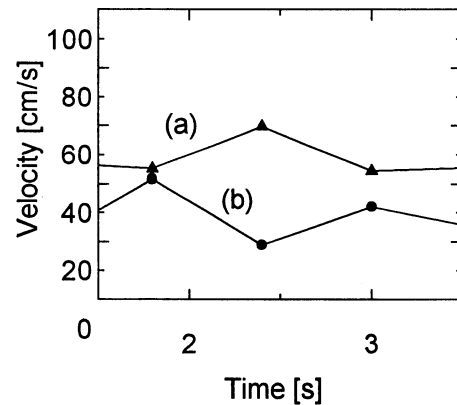
Fig.7 Velocity of walking human

should use under 0.5 [s] as a sampling time (max position change is 0.5[m] during the span). Our system outputs human head position asynchronously and the sampling time is less than 50ms. Thus we reconstruct human head at constant sampling time by software. We use the last 5 data from the current sampling time and calculate average of them. In Fig.7, there is average velocity of experimental results of walking human in the room. Data (a) is a fast walker, and (b) is a slow walker .

4.3 Direction of a Walking Human

Detection of human position and direction is an important function in Intelligent Space, because the system is able to assume what human is going to do with those data. If we can estimate the direction of a walking human and there is gate on his way, we can imagine what is going to happen. There are some possible application of using human direction and it might be useful. To detect human direction, the most simple way is to extract the human moving direction. Human sometimes look aside while he is walking, but his main concern is almost in front of him. With human direction the system selects his expected by motion from some candidates. Thus the system should have environment information such as desks, PCs, monitors, doors and etc. This method provides simple and useful artificial inference.

4.4 Personal Difference Detection

Individual Separation is important when there are more than one people in the room. There are some individual differences like height, walk style, actions of arms and etc. With a simple calculation the system was able to estimate human heights [Fig.8]. The output is quite robust of and it could find out humans height except someone who is walking strangely. There are 4 sample people who are from 165 [cm] to 175 [cm] tall.
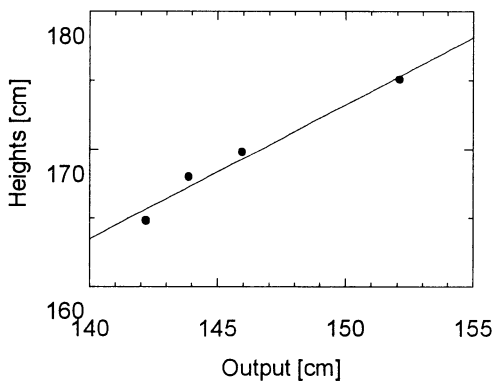
Fig.8 Heights of Individuals

We generate the equation (1) applying first order approximation.

$$H = 0.979 \times Z + 27.20 \qquad (1)$$

H: Heights of Human [cm], Z: Height of Head [cm]

The difference between output and humans height is 27.20 [cm]. It depends strongly on camera calibrations and location of the face. With this data the system recognizes individuals. We will have to analyze the data and apply some algorithm to recognize individuals as an future work.

4.5 Basic Action Analysis

In order to estimate human motion, we assume that human motion consists of some basic actions. We analyze some simple motion such as walking and etc. and find out some meaningful results. The way to analyze motion is to calculate velocity, acceleration and spectral amplitude of head position changes with offline.

Fig9 shows results when a human sits down. In A, solid line insists absolute velocity of x and y and dashed line is velocity of z. In B, solid line is acceleration of x and y and dashed line is that of z. From 13[sec] to 16[sec], a human sits down and from 20[sec] to 22[sec], he stands up. When sitting down, the change of z velocity leads the change of x, y velocity. The order of changes is opposite when standing.

Fig.10 is results of tracking when human walks slowly. 1, 2, 3 indicate the position changes of x, y, z directions. 4, 5, 6 indicate spectral amplitude of x, y, z directions. A is a result when a human walks along y direction and it outputs some feature of x directional rolling. He walked with obvious rolling motion intentionally. B is a result when a human walks along x direction and it outputs y directional rolling. In this time he walked slowly and naturally with thinking something. Each data has rolling peaks at 0.3 [Hz]. This is same as the timing when human moves his legs and the amplitudes of rolling are linear to width of rolling.

From now we will continue our work to analyze simple motion and to cluster basic actions. Currently we analyze data in off line, thus we combine an analyzing algorithm with 3D tracking vision systems and make a new function of detecting a basic action in real-time.

5. Conclusion

We have proposed a system that is capable of recognize a human motion from the movements of people in the room that works robustly in a complex

real world environment.

There are still some problems, for example individual recognition when some people walk at the same time. We need to apply more detailed human model for a individual separation.

A more general question is what a room can learn from looking at its inside. Future work of this project is the recognition of human's intention and inside statement from looking at their motions.

References

[1] J. H. Lee, N. Ando, H. Hashimoto, "Mobile Robot Architecture in Intelligent Space", in Journal of Robotics and Mechatronics Vol.11 No.2, 1999

[2] Takahiro Yamaguchi, Joo-Ho Lee, Hideki Hashimoto: "Human Motion Recognition with Vision Data", Proceedings of the 16th Annual Conference of the Robotics Society of Japan, Hujisawa, 1999.9

[3] K. Nakatsukasa, H. Hashimoto, F. Harashima, "Detection of Human's Confusion from Their Unconscious Action with Neural Network", Int. Conf. on Artificial Life and Robot, 1998

[4] H. Tsutsui et al.: Optical Flow-Based Person Tracking by Multiple Cameras, Proc. Workshop on Machine Vision Applications, pp. 418-421, 1998

[5] T Watanabe and M. Yachida: "Real Time Gesture Recognition Using Eigenspace from Multi Input Image Sequences", FGRC98, pp.428-433, 1998

[6] G. Appenzeller, J. H. Lee and H. Hashimoto, "Building Topological Maps by Looking at People: An Example of Cooperation between Intelligent Spaces and Robots", in Proceedings of IROS97, pp. 1326-1333, Sept. 1997
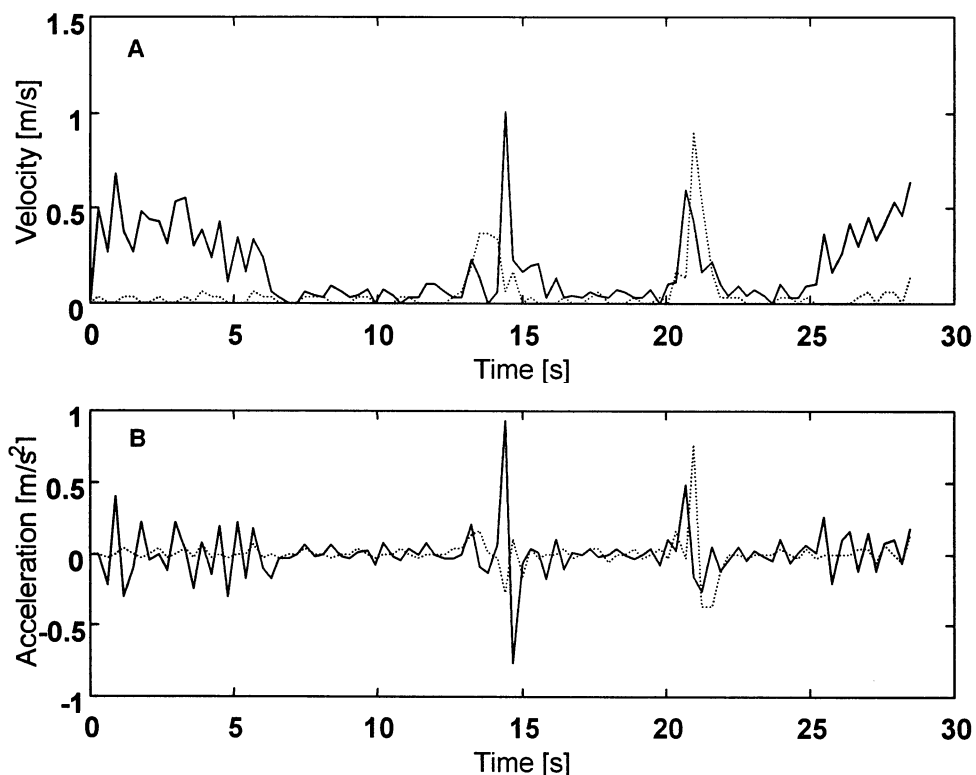
Fig.9 characteristics when sitting down
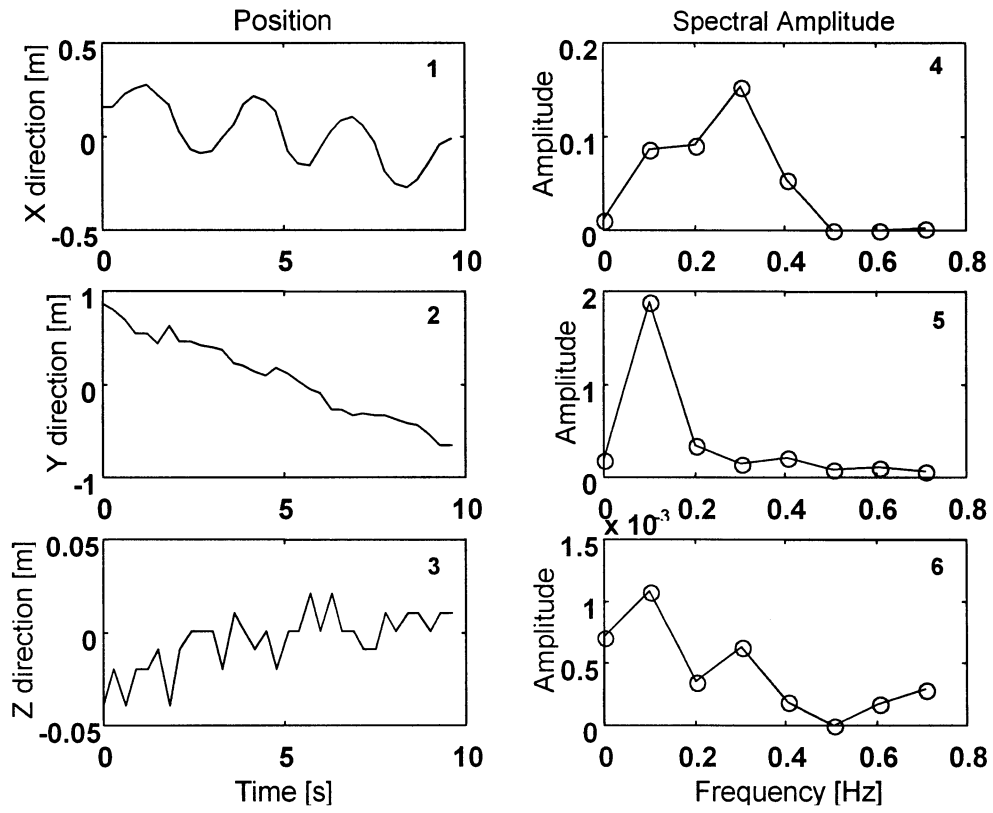
(a) intentional motion
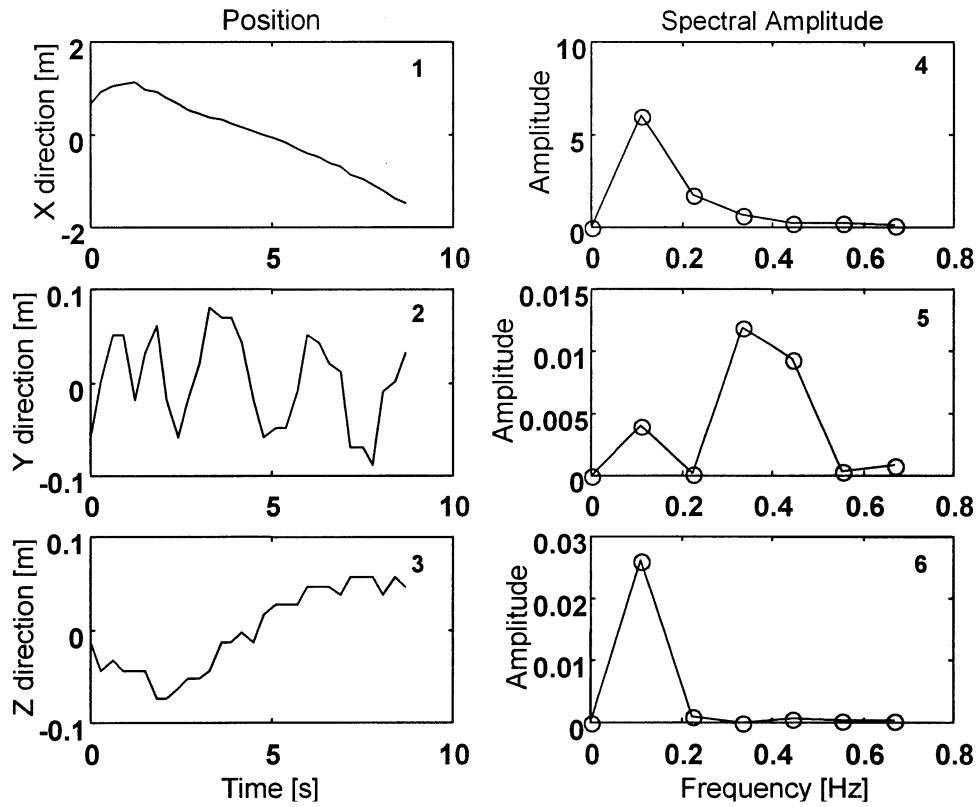
(b) unconscious motion

Fig.10 Rolling when walking slowly

Realization of a Mood Congruency Effects
based on a Model of Knowledge, Emotion and Intention

Tomomi HASHIMOTO
tomomi@nasu-u.ac.jp

Department of urban economics, Nasu University,
kanosaki 131, kuroiso-shi, 329-3121, JAPAN

Abstract   This paper, describes an experimental emotion model having mood congruency effects. An agent chooses sensor information and its own behavior according to its emotions.

Key words: Emotional computing, Robot, FCMs, FAMOUS

1. Introduction

The aged population is increasing, and the supporting younger population must provide assistance to handicapped people (a welfare support system). This system is outlined in **Fig. 1**, where handicapped people are supported by care-providing agents. The person orders an agent directly by gesture, and the instruction is input to the agent as sensor information. The agent recognizes movements such as "He points to a window" and "He walks" from the sensor information input and surmises the intention of those movements, for example, "He wants me to open a window" and "He wants me to assist in him walking". If then takes action such as "I will open a window", "I will assist walking". If the agent only has knowledge such as the "recognition of instruction and inferring the intention of movement" and he cannot instruct the agent, then the agent does not start movement. To produce the welfare support system, the agent needs to move automatically without instruction. For example, the agent should infer "He has a stomachache" and take the action to "inform an other person ", when he rubs his stomach by hand.

Hashimoto proposed a model of knowledge, emotion and intention [1]. Its structure adds "emotion" and "intention" to Rasmussen model (hierarchical knowledge construction such as skill, rule and knowledge) [2]. The model is an autonomous control mechanism for agents using Fuzzy
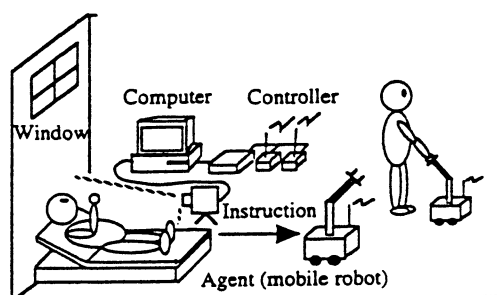
Fig. 1 Outline of the system providing physical assistance to handicapped people

Cognitive Maps (FCMs) [3]. But, only emotion patterns set beforehand appeared because the weights of the FCMs are fixed in the mode. Therefore, mood congruency effects were not expressed such as "when a feeling was good, then the information was positive", or "the information was negative when a feeling was bad".

On the other hand, Omori proposed PATON, which is a control system with a neural network [4]. PATON has mood congruency effects; even if it has similar inputs, it can determine by associative reasoning based on the situation.

In this paper, I propose the k.e.i. system, which incorporates a mechanism that
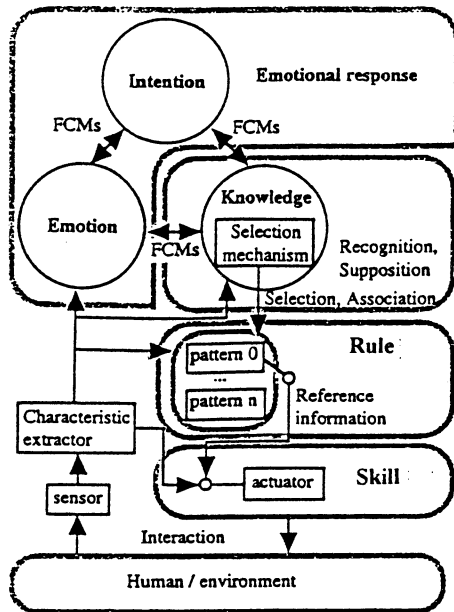
Fig. 2 Construction of engineering model for knowledge, emotion and intention. of agent

Fig. 3 Simple FCM construction
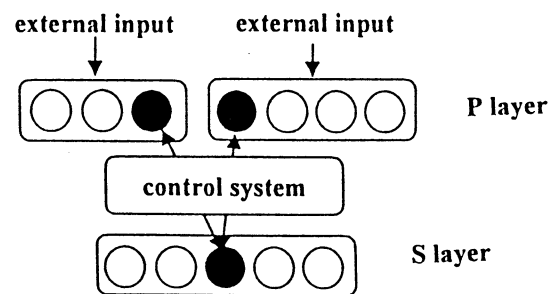
Fig. 4 Basic structure of PATON model (attention FAMOUS)

lets the above-mentioned model of knowledge, emotion and intention express situational dependency.

2. Trial implementation of engineering model for knowledge, emotion and intention

2.1 Hierarchical knowledge based on Rasmussen model

The Rasmussen model including the emotion and intention concept is an engineering model of knowledge, emotion and intention for the agent (**Fig. 2**). The agent converts information such as gestures and environmental data into characteristic information through a characteristic extractor. The characteristic information is sent to "knowledge", "rule" and "emotion". The agent recognizes gestures and the environment from the characteristic information, and surmises the intentions of the gestures and environment, based on its recognition results in "knowledge". The agent also decides the driving pattern of a motor in a "rule", by a selection mechanism in "knowledge" or associative reasoning between "knowledge" and "rule". Finally, the agent acts on the pattern. The "knowledge" is the recognition of gestures and the

environment, the discerning of intention from this recognition, a selection mechanism, and an association reasoning method. The "emotion" is the feeling of the agent toward the gestures and environment, and the "intention" is the purpose of the agent's movement. The agent works automatically by using a cause-and-effect relationship (emotional response) among "knowledge", "emotion" and "intention".

2.2 Emotional oriented interface using FCMs

A simple FCM model is shown in **Fig. 3**. The nodes C1, C2, ..., Cn Cn+1 are conception nodes such as "anxiety", "fright", "approach" and "leave" are conception grade of the node is a fuzzy membership grade, and the grade is between 0 and 1 such that 0.2 expresses a little, 0.5 normal and 0.9 is a lot. The weights w1, w2, ..., wn represent the grade of the cause-and-effect relationship between cause nodes (C1(t), C2(t), ..., Cn(t)) and the effect node (Cn+1(t+1)).

2.3 Structure of PATON and FAMOUS

PATON is a kind of associative memory that has a similar structure to the brain's macroscopic memory system (**Fig. 4**). It has a pattern layer P and symbol layer S, and a control system. In the PATON model, an outer world event is represented by the pair of a P layer attribute set and the corresponding symbol neuron. When the model recollects an event, the P and S layers are activated at the same time depending on the attention vector that selects attribute areas and symbols that can be activated. In this study, I built a PATON structure by using the attention FAMOUS, which was proposed by Yamaguchi [5].

3. Application to a mobile robot with mood congruency effects

The knowledge construction of **Fig. 3** is produced on a computer, suitable sensor information pattern is input to the construction, and the psychology variation (internal condition of agent shown with fuzzy perception manufacturing auto protocol) of the model is simulated. The agent has a voltage sensor, which is an internal sensor, and a CCD camera, which is an external sensor. Furthermore, it has PATON (attention FAMOUS) in "intention", and feelings which are two dimensions (excellent - indifferent in "emotion", and FCMs among "knowledge", "emotion" and "intention" each course.

The agent behaves in the following manner. First, it determines the fuzzy membership grade of the external environment from the sensor information (characteristic information), such as "power source voltage is 0.2" and "light intensity is 0.5". Next, it determines its own emotion from the FCMs, such as "excellent is 0.8" and "shrug is 0.4". The attention of FAMOUS (sensor selector) in "knowledge" changes to recognize the external environment if the emotion is relatively; otherwise, it excellent changes to recognize the internal environment. By using FAMOUS, the agent recognizes its internal condition such as "I (agent) am hungry" or "I am full", and recognizes the external environment such
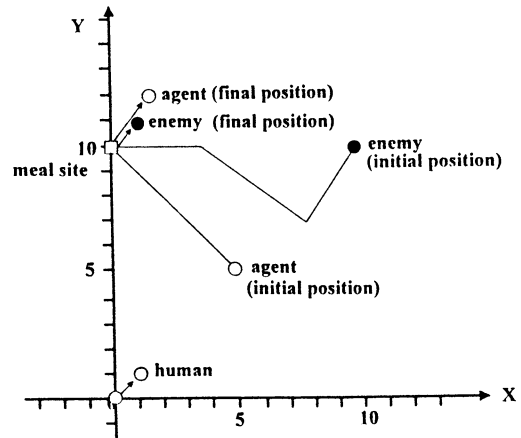
Fig. 5 Results of simulation (movement trace)
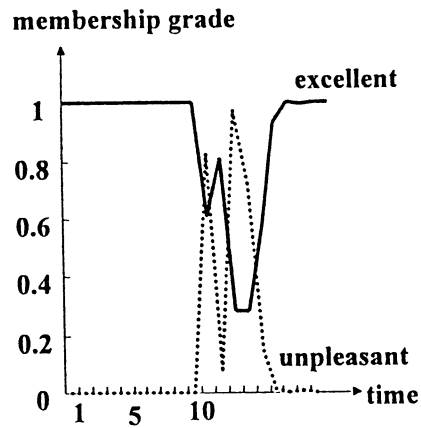
Fig. 6 Results of simulation (emotion)

as the presence of an "enemy" "meal site" and "persons". The agent determines the "intention" of the behavior such as "I approach the person" "I escape from an enemy" from the recognition result of "knowledge", and by the condition of "emotion". Finally, the agent determines a behavior pattern based on the situation, such as "behavior when the agent meets a person in the meal site" or "behavior when the agent meets an enemy in the meal site".

Figure 5 shows an agent's movement trace, and **figure 6** shows the agent's emotion.

In first, the agent is in the center of map. Because "emotion" is excellent, the agent recognizes the external environment, and it acts to take a meal. Emotion becomes unpleasant when the agent recognizes an

enemy at the time index of 8 (**Fig. 6**). The agent changes its attention to sense only the internal environment, to reduce psychological load. Therefore, the agent cannot recognize the external environment at all at time 9 (the agent cannot see the enemy, but is able to look it). The agent recognizes the external environment and evades an enemy at time 10, because it is feeling excellent.

Based on the principle of self-preservation, the agent takes evasive action if an enemy draws near. But it modifies its behavior based on situational dependency, such as taking evasive action at time 10 but not at time 9.

4. Conclusion

In this paper, I proposed k.e.i. system, which incorporates a mechanism that lets the model of knowledge, emotion and intention express situational dependency. I showed the system structure suggested by simulation. It incorporates the attention FAMOUS in a traditional model of knowledge, emotion and intention. The agent selects sensor information and a behavior pattern according to its emotion.

References

[1] Tomomi Hashimoto, Toru Yamaguchi and Jyuichi Miyamichi, "Emotion-oriented Man-machine Interface for Welfare Intelligent Robot", Journal of the Robotics Society of Japan, Vol.16, No.7, pp.993-1000, 1998 (in Japanese)

[2] Jens Rasmussen, "Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models", IEEE Transactions on Systems, Man, and Cybernetics, Vol.13, No.3, pp.257-266, 1983

[3] Bart Kosko, "Fuzzy cognitive maps," International Journal of Man-Machine Studies, Vol. 24, pp. 65-75, 1986

[4] Takashi Omori, "Theory of Symbol and Pattern Integration by Associative Memory with Attention", Proceedings of the 1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems, pp.438-445, 1996

[5] Toru Yamaguchi, "Fuzzy Associative Memory System", Journal of Japan Society for Fuzzy Theory and Systems, Vol.5, No.2, pp.245-260, 1993

Human Centered ITS

Toru YAMAGUCHI*, Hiroshi NITTA*, Kenichiro HIRAYAMA*,
Hideki HASHIMOTO** and Tomohiro TAKAGI***

*Department of Information Science, Faculty of Engineering, Utsunomiya University
7-1-2 Yoto, Utsunomiya-shi, Tochigi, 321-8585, JAPAN

**Institute of Industrial Science, University of Tokyo
7-22-1 Roppongi, Minato-ku, Tokyo, 106-8558, JAPAN

**Department of Computer Science, Meiji University
1-1-1 Higashi-mita, Tama-ku, Kawasaki, Kanagawa, 214-8571, JAPAN

yamachan@is.utsunomiya-u.ac.jp

Abstract

This paper's research aim is realizing an "Intelligent Space". This Intelligent Space is a platform when realizing multi agent systems. In order to build the multi agent systems, the most important element is a human interface technology in the Intelligent Space. It is important for this human interface technology to think about "universal design approach" and "cognitive science approach". These approaches are the convenient approach for all people: not only young people but also elderly people and handicapped people. In order to realize the human interface, this paper proposes an "augmented sensing". Human can feel invisible information to be visible information by using this augmented sensing.

This paper explains the augmented sensing in detail and show an effectiveness of the augmented sensing in the Intelligent Space.

Keywords : augmented sensing, Intelligent Space

1 Introduction

In the present when maintenance of an information infrastructure is advanced, the research about the intelligent man-machine system like a welfare support system is required. When such research progresses, the applications to the operation support for the automatic wheelchair in domestic and in medical institution, the operation support for the small electric vehicle in commuter systems, the operation support for Intelligent Transportation Systems (ITS), etc., are possible.

In order to build such function, we propose an "augmented sensing". Human can feel invisible information to be visible information by using this augmented sensing. The augmented sensing consists of two parts as follows:

(1) Re-composition of sensory information part:

Each agent integrates their own sensory information with the sensory information which other agents obtained and re-composes the information.

(2) Display of composed information output part:

The agent displays the camposed information.

This paper explains the augmented sensing in detail and show an effectiveness of the augmented sensing in an "Intelligent Space".

2 Space with distributed sensory intelligence: Intelligent Space

2.1 Intelligent Space and assistance to man

Intelligent Space[1][2] is the space which has distributed sensory intelligences (various sensors, such as cameras and microphones + intelligences) in the whole space. The sensory intelligences cooperate with each other autonomously, and the whole space have high intelligence.

Each intelligent agent in the Intelligent Space has sensory intelligence (sensor + intelligence). The intelligent agent has to operate, even if the outside environment changes. So it needs to switch its role autonomously (Section 2.3). The agent knows its role and can support man. At this time, each agent obtains the sensing information from multiple input attributes, such as cameras and microphones. In addition, it obtains the augmented information from the other agents. The agents can re-compose the whole space by using the augmented sensing information (Chapter 3). In addition, the agents need to display intuitively and intelligibly the augmented sensing information and the support information to man(Chapter 4).

In Intelligent Space (**Fig. 1**), each intelligent agent switches their role autonomously. Intelligent Space re-composes the whole space from each agent's sensing information, and returns intuitive and intelligible reactions to man. In this way, Intelligent Space is the space where

man and agents can act mutually.
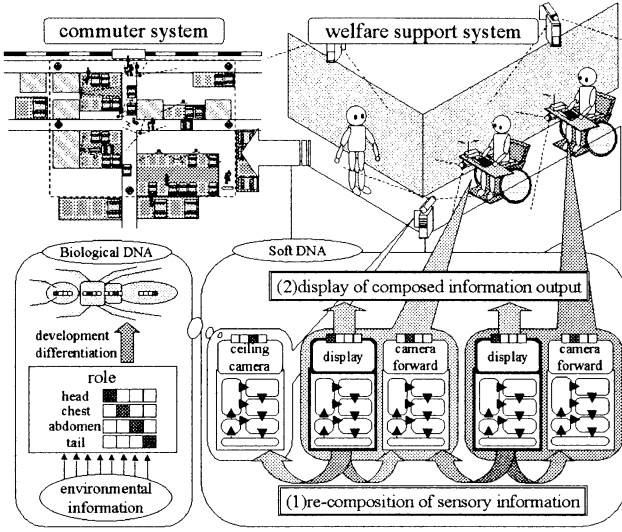
Figure 1: Intelligent Space

2.2 Intelligent agent model

As a model of the agent in Intelligent Space, we assume a hierarchical intelligent agent model which extends the hierarchical model of Rasmussen[3] (**Fig. 2**).
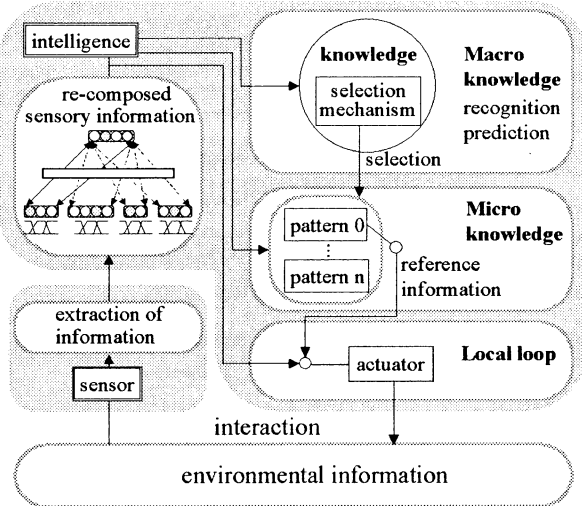
Figure 2: Hierarchical intelligent agent model

The hierarchical model of Rasmussen is known as an effective model which builds an intelligent model. The model divides the category about action of man into three hierarchical structures called macro knowledge, micro knowledge, and local loop. Macro knowledge (which is the highest order) evaluates, micro knowledge corrects, and the local loop (which is the lowest order) controls. The more the hierarchy becomes a high order, the more it treats abstract information.

2.3 Autonomous role switching by using Soft DNA

Each agent in Intelligent Space needs to switch their role autonomously and cooperates with each other according to the outside environment. **Fig. 3** shows the image of "Soft computing oriented Data driven fuNctional scheduling Architecture" (i.e., Soft DNA)[4] with the mechanism which switches role.
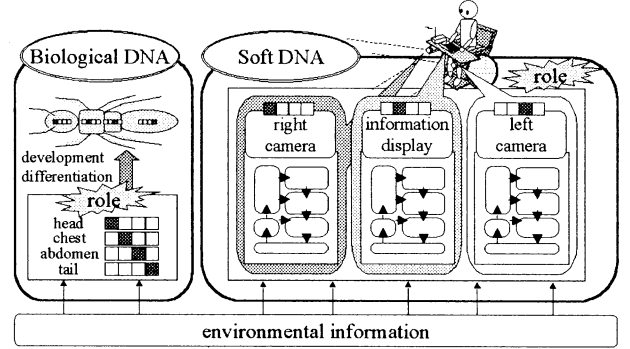
Figure 3: Biological DNA and Soft DNA

Soft DNA aims to imitate the idea of the developmental process, such as the body plans in actual life based on biological Deoxyribo Nucleic Acid (biological DNA). In biological DNA, genes called "Homeo box genes" dynamically control the body development of an individual in actual life based on the concentration of protein which is influenced by itself and the outside. Similarly, Soft DNA has Homeo box genes (such as information display, right camera, and left camera in **Fig. 3**), and collects them into one. Each agent has common Soft DNA and can know their role because of information of the outside environment. In this way, each agent in Intelligent Space can switch their role autonomously and cooperate with each other according to the outside environment.

3 Re-composition of augmented sensing information using "Attention"

According to the outside environment, each intelligent agent in Intelligent Space needs to switch their role autonomously and support man in cooperation with each other. And it is important for each agent to re-compose the whole space from augmented sensing information by using the information of the surrounding agents. Using "Attention", the agent determines which agent's information was observed at this time.

This chapter explains "Attention" first. In addition, it proposes a method of re-composition of the whole space from augmented sensing information by using the information that each distributed sensory intelligence got.

3.1 Attention

3.1.1 Neocortico-hippocampal model and Attention

A hippocampas is one of the memory systems of the brain in connection with cognition, consciousness, etc., of man. The neocortico-hippocampal model[5] in **Fig. 4** modeled the hippocampas.
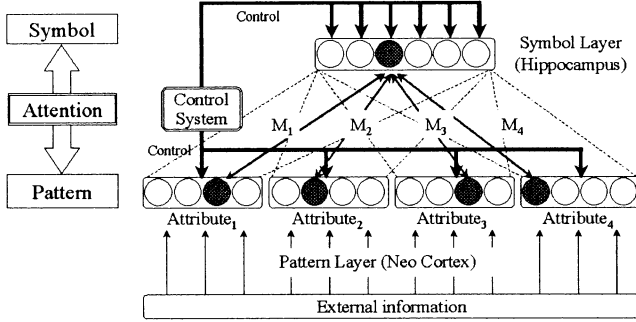
Figure 4: Neocortico-hippocampal model

The neocortico-hippocampal model is very effective in recognizing and learning using multiple elements.

This model consists of two layers: a pattern layer and a symbol layer. It is expressed in the pairs of the element expression of the pattern layer and the sign expression of the symbol layer. When certain things and concepts are recollected, the cell group of both the pattern layer and the symbol layer will be excited. At this time, an attention vector controls the pattern layer in order to use only a part of the element or checks retrieval of a part of the symbol layer. Although there are various functions in the attention, "Attention" in this paper is a function which controls the pattern layer and the symbol layer.

3.1.2 The role of Attention in an intelligent agent model

This subsection explains the role of Attention in an intelligent agent model by using **Fig. 4**. This figure is the part which integrates augmented sensing information and re-composes the whole space in **Fig. 2**. Specifically, each agent integrates multiple input attributes (such as their sensing information and other agents' sensing information) and can re-compose the whole space from augmented sensing information. Attention based on the neocortico-hippocampal model can obtain the integrated information by controlling the attention to the attributes observed. **Formula(1)** expresses how to control a attention vector by Attention based on the neocortico-hippocampal model.

$$M = attn_1 M_1 + attn_2 M_2 + \cdots + attn_n M_n$$
$$(attn_i \in [0, 1])$$ (1)

Here, $attn_1, attn_2, \cdots, attn_n$ are the variables which control attention by each attribute, $M_1, M_2, \cdots, M_n$ are individual associative matrices, respectively, and M is the associative matrix integrated by Attention.

In **Fig. 4**, in case multiple attribute patterns retrieve one symbol, Attention controls which attribute is observed. As a result, one result can be obtained. For example, suppose that there are $attribute_1$ and $attribute_2$. If we want to observe $attribute_1$ but do not want to observe $attribute_2$, we should just bring the value of $attn_1$ and $attn_2$ close to 1.0 and 0.0, respectively. In this way, Attention can control attention to each attribute by making the value of $attn_i$ into a value between 0 and 1. As a result, the agent can re-compose the whole space from the sensing information which each agent obtained.

3.2 Application to ITS
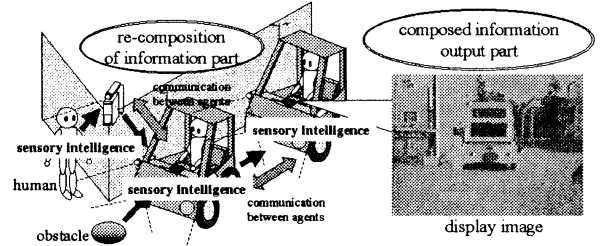
Figure 5: ITS system
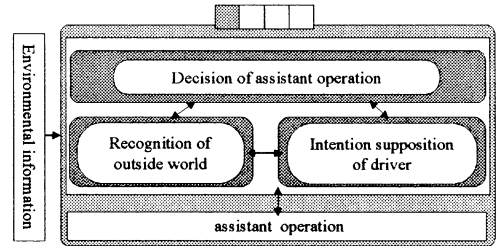
Figure 6: ITS car model

In Intelligent Space, **Fig. 5** shows a method of re-composition of the whole space from the augmented sensing information that each agent got using Attention.

The **Fig. 6** shows the ITS car model. ITS car has three function, recognition of outside world, intention supposition of driver and decision of assistant operation. This paper pick up recoginition of outside world, and realize by below method.

An car has a sensory intelligence (sensor + intelligence) and is equipped with a front camera and a display for information display. The person in the car can obtain information in front with the front camera. However, when the front is not visible because an obstacle, augmented sensing information about space from other agents' sensing information can re-compose information

from the front.

Image information obtained from the camera is sent to the computer, and the computer extracts the colors and motions by software processing. In this way, the information on the position and direction of a pedestrian can be obtained.

In addition, the agent can use this information with other agents. by sharing information that each agent got through the network. Using the sensing information obtained itself and by the surrounding agents, the agent can augment the sensing information about space by using Attention and can re-compose the whole space.

The picture on the right-hand side of **Fig. 5** shows a pedestrian's information for a back car.

3.2.1 Experiment on ball pursuit

4 Display of composed information output

In operation support of automatic wheelchair, small electric vehicle, and ITS vehicle, etc., there is a display function which displays intelligibly obtained sensing information and information about operation assistance to man as one of required functions.

In the display function, it is important that an agent urges to attention clearly to all people. In other words, a display based on intuition of man becomes important.
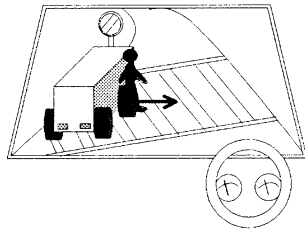
Figure 7: Head-up display

Fig. 7 shows a Head-Up Display. The Head-Up Display can project operation support information and augmented sensory information on front window.
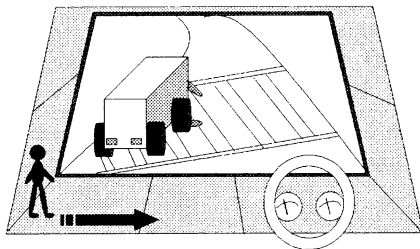
Figure 8: Displays around a front window

Fig. 8 shows displays around a front window. Even if it doesn't project on front window, the display can project operation support information and augmented sensory information to human.

In the future, not only ordinary display but also using both the Head-up Display and displays around the front window is possible.

We would like to advance the augmented sensing by considering as a work of the future.

5 Conclusions

In the Intelligent Space, a lot of agents with sensory intelligence are distributed. The agents switch their role dynamically and support man by cooperating with each other. At this time, it is important for the agents to use efficiently the information obtained from the distributed sensors. So each agent integrated their own sensory information with the sensory information which other agents got. We used the "Attention" as a mean to integrate the distributed sensory information.

In order to realize the human interface, we proposed the augmented sensing. Human can feel invisible information to be visible information by using the augmented sensing. The augmented sensing consists of two parts as follows:

(1) Re-composition of sensory information part
(2) Display of composed information output part

This paper explained the augmented sensing in detail and showed an effectiveness of the augmented sensing in the Intelligent Space.

References

[1] G. Appenzeller, J. Lee, H. Hashimoto, "Building Topological Maps by Looking at People: An Example of Cooperation between Intelligent Spaces and Robots," *IEEE/RSJ International Conference on Intelligent Robots and Systems, Grenoble France,* 1997.

[2] K. Hirayama, T. Arai, T. Yamaguchi, H. Hashimoto, "Human Motion Recognition on Intelligent Space," *Proceedings of the Electronics, Information and Systems Conference Electronics, Information and Systems Society, I.E.E. of Japan,* pp. 105-108, 1998. (in Japanese)

[3] J. Rasmussen, "Skills, Rules, and Knowledge," *Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models, IEEE Trans. on SMC,* Vol.13, pp. 257-266, 1993.

[4] M. Sato, Y. Wakamatsu, N. Kohata, T. Yamaguchi, "Dynamic Formation Generating in Multi-Agent using Soft DNA and Chaotic Evolutionary Computation," *14th Fuzzy System Symposium,* pp. 591-594, 1998. (in Japanese)

[5] T. Omori, "Computation Theory of Pattern and Symbol Interaction", *Journal of Japanese Neural Network Society,* vol. 3, No. 2, pp. 65-67, 1996. (in Japanese)

Diversity Oriented Evolutionary Parallel Computation based on Cultural Background

Naoki Kohata, Toru Yamaguchi, Takanobu Baba and Hideki Hashimoto*

Department of Information Science, Faculty of Engineering, Utsunomiya University
7-1-2 Youtou, Utsunomiya-shi, Tochigi-ken, 321-8585, Japan
kohata@sophy.is.utsunomiya-u.ac.jp
*University of Tokyo, Institute of Industrial Science
7-22-1 Roppongi, Minato-ku, Tokyo-to, 106-8558, Japan

Abstract

This paper proposes the generation of diverse intelligence on a chaotic evolutionary computation algorithm based on cultural background for such mobile-agent systems as Intelligent Transport Systems (ITS). This evolutionary computation is realized by applying chaotic retrieval and Soft DNA(Soft computing oriented Data driven fuNctional scheduling Architecture) on associative memories. We apply this evolutionary computation to ITS. Essentially, the process of this evolutionary computation is parallel processing. Therefore, we also propose its parallel processing algorithm. We show the usefulness of the proposed method by means of simulation experiment.

Figure 1. Evolutionary computation of intelligence by chaotic dynamics

1 Introduction

Recently, evolutionary computation models on Alife (Artificial life) have been researched by computer. Nowadays, its typical approach method is GA (Genetic Algorithm). However, it is certain that evolution of actual life is not such simple processes as conventional GA. Above all, we think there are not only genetic factors but also other factors (e.g., cultural factor) in evolutionary process of brain or its intelligence. In addition, it is said that evolution is irreversible process which does not enable the life to become again the exactly same life as it used to be. It seems to us that there is chaos in this complexity of evolution.

Therefore, we propose evolutionary computation of intelligence by chaotic dynamics and Soft DNA (Soft computing oriented Data driven fuNctional scheduling Architecture) as shown in **Fig.1** [1]. We explain this Soft DNA in detail later. This evolutionary computation flow contains the generation of diverse intelligence based on a kind of cultural background. **Fig.2** is the image of parallel evolutionary computation which introduced cultural background. We explain the diversity simulation experiments later.

In this paper, we use a method which enables efficient work as a group by dynamically selecting the role allotment in the agent group and the role. The method is "Soft computing oriented Data driven fuNctional

scheduling Architecture (i.e., Soft DNA)", which is a new algorithm that agent voluntarily selects function by using environmental information. We also propose chaotic evolutionary computation on soft DNA. Moreover, we introduce a kind of cultural background into the proposed evolutionary computation in order to generate the diversity of intelligence.

Agent's control block is constructed by Fuzzy Associative Memory Organizing Units System (FAMOUS) [2] and Conceptual Fuzzy Sets (CFS) [3]. The effectiveness of the our method is verified by the simulation experiment by applying to ITS.

2 Cooperative Works on Mobile-Agent System and its Evolutionary Computation

When cooperatively operating by several agents, we propose the idea to make each agent allot work. For instance, when working to assemble a certain product, agents which carry parts, agents which assemble carried parts and agents which complete the product is necessary. Each agent has its own post. But when they move, some agents part from their own post or go
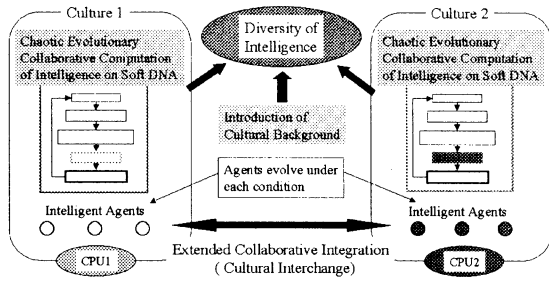
Figure 2. Parallel evolutionary computation image introduced cultural background
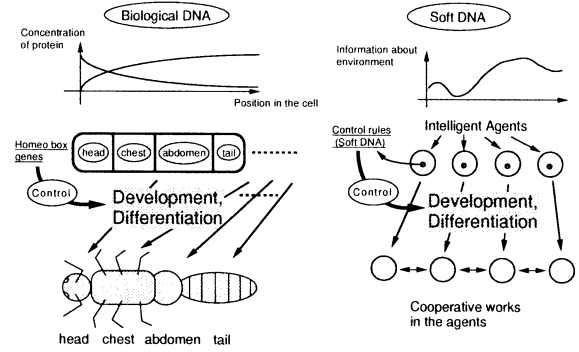
Figure 3. Biological DNA and Soft DNA.

into to other agents' posts. For that case, agents exchange roles mutually. We explain the method called Soft DNA and its Evolutionary Computation.

2.1 Soft DNA and its Evolutionary Computation

In a system in which a large number of agents work, it is good for all agents to share roles. To produce cooperative works, we have a proposal of a new method. We call this new method "Soft DNA (Soft computing oriented Data driven fuNctional scheduling Architecture)". Soft DNA(**Fig. 3**) aims to imitate the image of the developmental process, such as the body plans in actual life based on biological DNA (DeoxyriboNucleic Acid).

In biological DNA, the genes called "Homeo box genes" dynamically control the body development of an individual in actual life based on the concentrations of proteins in cells. The control architecture called "soft DNA" dynamically controls the development of intelligence in each agent based on environmental information, in order to achieve dynamic cooperation. Biological DNA has sets of genes that are related to each body part such as the head, chest, abdomen, and tail as shown in **Fig. 3**. These sets of genes are each called a homeo box. Similarly, soft DNA has boxes of intelligence (made by soft computing, i.e., associative memories, neural networks, fuzzy logic, chaos, and so on) that are related to various environments, a suitable box of intelligence is developed according to the environmental information available as shown in **Fig. 3**. All agent robots have the same soft DNA and can select their own roles dynamically.

The proposed soft DNA consists of some boxes which are made by associative memory system. We use chaotic retrieval [4] on associative memory to carry out evolutionary computation on soft DNA. We explain the systems based on associative memory which are used in our research in the next section.

2.2 FAMOUS and CFS

Agent's control block is constructed by following FAMOUS and CFS.
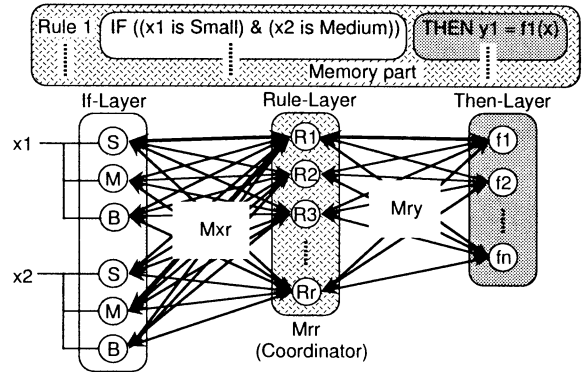
Figure 4. Fuzzy Associative Memory Organizing Units System.

The FAMOUS(**Fig. 4**) [2] is a system that realizes fuzzy associative inference by constructing fuzzy knowledge on associative memory. The FAMOUS adopts BAM(Bidirectional Associative Memory) [5] which enables bidirectional retrieval.

Conceptual Fuzzy Sets (CFS) [3] are new fuzzy sets that conform to Wittgenstein's ideas [6] on the meanings of concepts. A CFS is achieved as an associative memory, combining a long-term memory and a short-term memory, thereby reducing the complexity of knowledge representation.

CFS represents the meanings of concepts in multiple layers. The meaning of a concept is translated into an expression indicated by the distribution of activation in each layer. Propagations arise from the activation of the concept. In contrast, the activation of a lower concept determines the activation of an upper concept, it corresponds to recognition or understanding.

2.3 Cooperative Works for ITS

ITS are the systems which control the distance between cars and make the amount of traffic increase. In the research field about ITS in recent years, the automatic charge system of a highway, the auto-cruise system , and so on, are studied. But we aim to achieve car group running, which is said "platoon running".

The control block for platoon running in this paper is shown in **Fig. 5**.
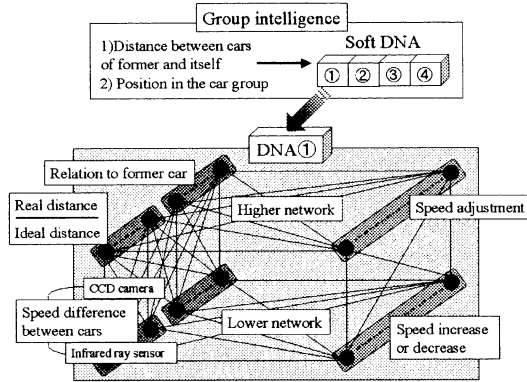
Figure 5. Agent Control Block for Platoon Running.

3 Simulation Experiments

3.1 Experimental Results about Soft DNA on ITS
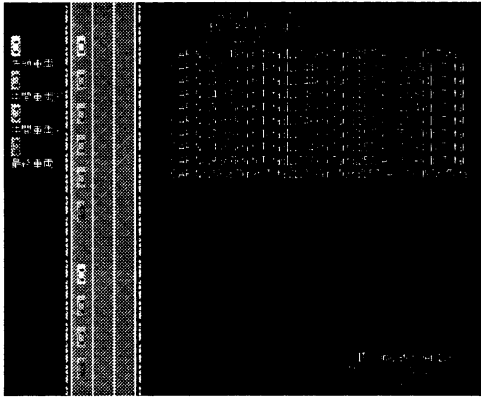
Figure 6. ITS Simulation Image.

Fig. 6 is an image which performed the platoon running by two or more vehicles. We applied soft DNA, which regulated the control knowledge of four roles of a head, middle 1, middle 2, and a tail. In order to verify the usefulness of soft DNA in this paper, we compare the case using soft DNA with the case not

Table 1. Simulation results of ITS.

(1) Using soft DNA	
Classification between vehicles with each role	Distance error (meter / sec)
Head - Middle 1	1.29
Middle 1 - Middle 2	1.25
Middle 2 - Tail	1.36
(2) Not using soft DNA	
—	4.04

using soft DNA. In the case not using soft DNA, all cars run in the same control knowledge. The triggers to switch role are a position of car in the car group and a distance between cars. In the simulation, the scope of a sensor was set at 200m. It is assumed that the one car can obtain information about another cars in the distance within 200 meters of the back and forth. Moreover the car can obtain other car's information at the same time. Thereby, several cars included in a car group can obtain common information. The group intelligence determines own role according to the above environmental information.

The total of the error of the distance between the distance between ideal cars and real car was taken and the averages were compared. **Table 1** shows the simulation results.

The average of the vehicular gap error the case not using soft DNA is 4.04m. When soft DNA was applied, the average of the vehicular gap error improved about 1/4 by chaotic evolutionary computation on soft DNA.

3.2 Experimental Results about ITS Diversity Simulation

In platoon running, if all platoons or all cars run based on the exactly same control knowledge, the traffic system may be a failure as a whole. Therefore, it seems that each platoon or each cars needs to have the fluctuations of its intelligence in order to do the different movement from the others.

From the above viewpoint, we propose an evolutionary parallel computation which generates the diversity of intelligence in agent group. In the proposed computation model, we introduce a kind of cultural background into our evolutionary computation in order to generate diversity as shown in **Fig. 2**. We are verifying an effect of the diversity by means of ITS basic simulation now. **Table 2** shows the whole table image of the expected simulation result.

First of all, we can show the result on traffic efficiency in **Table 3** from ITS distance control simulation in the previous section. ITS distance control result is related with the rectangular part by thick solid line in the whole table as shown in this table. The result shows that our chaotic evolutionary computation on

Table 2. Whole table image of ITS diversity simulation experiments.

Percentage of ITS vehicles (%)	0	The same rule 40	80	100
		Different rule 40	80	100
Traffic efficiency at normal time	✕	△	○	◎
		△	□	○
Degree of not coming to a deadlock at accident time	○	△	✕	✕✕
		○	○	◎

◎ : Very Good, ○ : Good, □ : A Little Good
△ : A Little Bad, ✕ : Bad, ✕✕ : Very Bad

Table 3. Simulation result (1): Traffic efficiency (Distance control)

Percentage of ITS vehicles (%)	0	The same rule(with no diversity) 40	80	100	· Average error of ITS distance control
		Different rule(with diversity) 40	80	100	Not using Soft DNA : 4.04 m Using Soft DNA : 1.30 m Evolved Soft DNA : 0.87 m
Traffic efficiency at normal time	✕	△	○	◎	Average error improved about 1/4 by evolved Soft DNA
		△	□	○	
Degree of not coming to a deadlock at accident time	○	△	✕	✕✕	◎ : Very Good, ○ : Good, □ : A Little Good, △ : A Little Bad, ✕ : Bad, ✕✕ : Very Bad
		○	○	◎	

All cars run in the same way   Each car runs in the different way by diversity of rules

If complete rules for all situations are not described, the system may come to a deadlock easily

As the result, the system may be able to avoid the obstacles, not coming to a deadlock easily

Figure 7. Diversity simulation experiment on an obstacle avoidance

Table 4. Simulation result (2): Degree of not coming to a deadlock (Diversity effect on an obstacle avoidance)

Percentage of ITS vehicles (%)	0	The same rule(with no diversity) 40	80	100	◎ : Very Good, ○ : Good, □ : A Little Good, △ : A Little Bad, ✕ : Bad, ✕✕ : Very Bad
		Different rule(with diversity) 40	80	100	
Traffic efficiency at normal time	✕	△	○	◎	Average error improved about 1/4 by evolved Soft DNA
		△	□	○	
Degree of not coming to a deadlock at accident time	○	△	✕	✕✕	: 72 times / 270 times (Success No. / Total No.) : 132 times / 270 times
		○	○	◎	

soft DNA can improve the traffic efficiency.

Next, we verified the effect of diversity by means of basic simulation on an obstacle avoidance (**Fig. 7**). We show the result on the degree of not coming to a deadlock from this simulation in **Table 4** In this case, "coming to a deadlock" means that each car cannot avoid the obstacle (i.e., cannot change lanes smoothly) and a traffic jam arises finally. These numbers of times not coming to a deadlock, i.e., numbers of success times are shown in the rectangular part by thick solid line in this **Table 4**. This result shows that we may be able to realize the systems which don't come to a deadlock easily by diversity of intelligence(different rules) in comparison with the same rule.

4 Conclusions

This paper proposed the generation of diverse intelligence on a chaotic evolutionary computation algorithm based on cultural background. This evolutionary computation was realized by applying chaotic retrieval and Soft DNA(Soft computing oriented Data driven fuNctional scheduling Architecture) on associative memories. We applied this evolutionary computation to ITS. We showed the usefulness of the proposed method by means of simulation experiment.

References

[1] N.Kohata, T.Yamaguchi, T.Baba, H.Hashimoto: Chaotic Evolutionary Parallel Computation on Intelligent Agents, Journal of Robotics and Mechatronics, Vol.10, No.5, pp.424-430 (1998).

[2] T.Yamaguchi: Fuzzy Associative Memory System, Journal of Japan Society for Fuzzy Theory and Systems, Vol.5, No.2, pp.245-260 (1993).

[3] T.Takagi, T.Yamaguchi and M.Sugeno: Conceptual Fuzzy Sets, International Fuzzy Engineering Symposium'91 (IFES'91), Vol.2, pp.261-272 (1991).

[4] J.Tani: Proposal of Chaotic Steepest Descent Method for Neural Networks and Analysis of Their Dynamics, Trans.IEICE, Vol. J74-A, No.8, pp1208-1215 (1991).

[5] B.Kosko: Adaptive Bidirectional Assosiative Memories, Applied Optics, Vol.26, No.23, 4947-4960 (1987).

[6] Wittgenstein: Philosophical Investigations, Basil Blackwell, Oxford, (1953).

Visualization of Patterns and Construction of Fuzzy Inference System for Integration of Patterns and Symbols

Y. Hattori* T. Furuhashi**

Dept. of Information Electronics, Graduate School of Engineering, Nagoya University

Furo-cho, Chikusa-ku, Nagoya 464-8603 Japan

Email: {hattori*, furu**}@bioele.nuee.nagoya-u.ac.jp

Abstract

Studies on integration of patterns and symbols have attracted many researchers. In this paper, patterns are feature vectors in a feature space and symbols are labels of sets of patterns. The integration of patterns and symbols is to treat pattern-symbol pairs for information processing.

Fuzzy inference is an integration of patterns and symbols by treating the pattern sets as fuzzy sets and the symbols as the labels of fuzzy sets. Fuzzy inference is very useful for extraction and incorporation of humans' knowledge. However, relationships between symbols and multi-dimensional pattern sets are hard for us to grasp. This paper proposes a fuzzy inference system with a nonlinear projector for dimensionality reduction to visualize relationships between the symbols and the multi-dimensional pattern sets. The proposed system is a good tool for interactive knowledge acquisition / incorporation.

Keywords:

Patterns and Symbols, Visualization, Dimensionality Reduction

1 Introduction

Studies on integration of patterns and symbols have attracted many researchers[1]-[3]. The definitions of patterns and symbols, and their integration differ somewhat from a researcher to another.

This paper discusses the definition of patterns and symbols by which patterns are feature vectors in a feature space and symbols are labels of sets of patterns. The integration of patterns and symbols is to treat pattern-symbol pairs for information processing.

Fuzzy inference is a kind of integration of patterns and symbols. Fuzzy inference treats pattern-symbol pairs in the form of fuzzy sets. The fuzzy inference maps pattern sets to another pattern sets using a map from symbols to symbols. One of the advantages of the fuzzy inference is that the map using the pattern-symbol pairs is very useful for extraction and incorporation of humans' knowledge.

However, if input patterns are in a multi-dimensional space, the production rules become complicated and hard for us to grasp because the relationships between the multi-dimensional patterns and the production rules are not visible. Visualization of the pattern sets by dimensionality reduction is one of the solutions to this problem. The production rules that are assigned to the visualized abstract patterns are simple.

This paper proposes a fuzzy inference system with a nonlinear projector for dimensionality reduction. In this system, multi-dimensional pattern sets are visualized in a two-dimensional space by the nonlinear projector, and then, the visualized abstract patterns are inputted into the fuzzy inference system. The proposed system can incorporate humans' symbolic knowledge through the fuzzy inference system and this knowledge is used for the adaptation of the nonlinear projector. The proposed system is also a good tool to acquire simple symbolic knowledge from multi-dimensional data. The incorporation and acquisition of knowledge can be proceeded interactively in this system. The two-dimensional patterns are visible, and the behavior of the system becomes easy for us to grasp. In this paper, the proposed system is applied to a controller for an autonomous mobile robot with multiple distance sensors.

2 Visualization of Patterns

2.1 Dimensionality Reduction and Fuzzy Inference System

Fuzzy inference is a kind of integration of patterns and symbols. In a fuzzy inference system, a map from a pattern to another pattern is realized by a procedure of maps from a pattern to a symbol, from the symbol to another symbol, and then from the symbol to a pattern. The map from symbols to symbols consists

of a set of production rules. The following rule is an example in the case where a 2-dimensional pattern x is the input and y is the output.

- If x_1 is Big, x_2 is Small then y is Big.

"Small" and "Big" are symbols. These symbols are labels assigned to fuzzy sets (pattern sets). This linguistic expression is easy for us to grasp. We can incorporate our knowledge into the system utilizing the pattern-symbol pairs.

However, if the dimension of the input space is large, the rules become complicated. Extraction and incorporation of humans' knowledge are hard in this case.

Visualization of the pattern sets by dimensionality reduction is one of the solutions to this problem. The production rules using symbols assigned to the visualized abstract pattern sets are simple. It is assumed that the p-dimensional pattern $x = (x_1, x_2, \cdots, x_p)$ is projected into a 2-dimensional pattern $x' = (x'_1, x'_2)$ by the nonlinear projector π.

$$\pi : x \to x'. \tag{1}$$

Then, fuzzy inference is done by using the abstract pattern x'. The following rule is an example in the fuzzy inference using the abstract pattern sets.

- If x'_1 is Big and x'_2 is Medium, then y is Big.

We can grasp the behavior of this fuzzy inference system by observing the abstract pattern sets. The problem here is how to define the projector π for dimensionality reduction.

2.2 Proposed System

Fig.1 shows a basic configuration of the proposed system. This system consists of a nonlinear projector which is a 3-layered network and a fuzzy inference system which is a fuzzy neural network (FNN). When an unknown p-dimensional pattern $x^* = (x_1^*, x_2^*, \cdots, x_p^*)$ is inputted into the input layer in the nonlinear projector, a 2-dimensional abstract pattern $x'^* = (x_1'^*, x_2'^*)$ is outputted in the abstract layer, and then, the FNN infers an output pattern y^* according to the abstract pattern x'^*.

2.2.1 3-layered Network

In the 3-layered network, inputs and outputs of units in the hidden layer and the output layer are defined by

$$\begin{cases} I_j^{(n)} = \Sigma_k w_{jk}^{(n)} O_k^{(n-1)} \\ O_j^{(n)} = f(I_j^{(n)}) \end{cases} \tag{2}$$
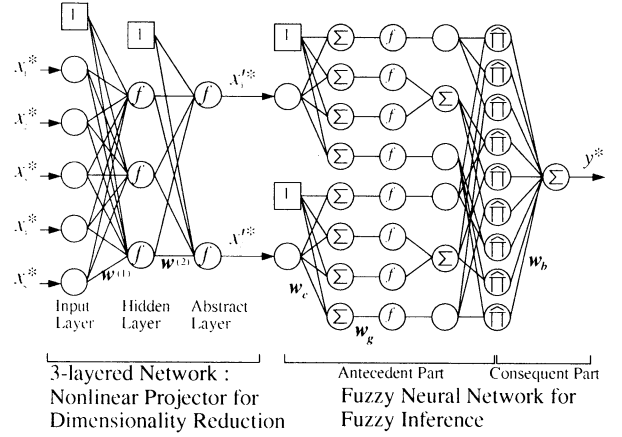
Figure 1: Proposed System

where $I_j^{(n)}$ is the input into the j-th unit in the n-th layer, $O_j^{(n)}$ is the output in the j-th unit in the n-th layer, $w_{jk}^{(n)}$ is the connection weight between the k-th unit in the $(n-1)$-th layer and the j-th unit in the n-th layer, and $f(\cdot)$ is a sigmoid function given by

$$f(x) = \frac{1.0}{1.0 + \exp(-x)}. \tag{3}$$

2.2.2 Fuzzy Neural Network

In the fuzzy neural network, the connection weights w_g and w_c determine the shape and the central position of a membership function in the antecedent part, respectively. The i-th membership grade $\mu_k^i(x_k'^*)$ is defined by the following equation:

$$\mu_k^i(x_k'^*) = f(w_{gn}(x_k'^* + w_{cn})) - f(w_{gp}(x_k'^* + w_{cp})) \tag{4}$$

where $f(\cdot)$ is a sigmoid function. In the consequent part, a normalized activation value $O_j^{(\hat{\Pi})}$ of the j-th rule is calculated according to the membership grades of the j-th rule as follows:

$$O_j^{(\hat{\Pi})} = \frac{\Pi_{k=1}^2 \mu_k^{(j)}(x_k'^*)}{\Sigma_{p=1}^N \left\{ \Pi_{k=1}^2 \mu_k^{(p)}(x_k'^*) \right\}} \tag{5}$$

where N is the number of rules. Output y^* of the FNN is given by the following equation:

$$y^* = \Sigma_{j=1}^N w_{bj} O_j^{(\hat{\Pi})} \tag{6}$$

where w_{bj} is the connection weight, corresponding to the singleton in the consequent part of the j-th rule.

2.2.3 Training of Network

The proposed system is trained by the back propagation method. Initial connection weights in the 3-layered network in the projector are very important for the training of the FNN, because distribution of the projected abstract patterns affects the design of the fuzzy inference system greatly.

In this paper, a 5-layered network shown in Fig.2 is used to initialize the connection weights in the 3-layered network of the proposed system. The 3-layered network of the proposed system in Fig.1 is installed into the first three layers of the 5-layered network as shown in Fig.2. This network acquires a map from p-dimensional input patterns $x = (x_1, x_2, \cdots, x_p)$ to input-output patterns $z = (x_1, x_2, \cdots, x_p, y)$ by the back propagation method. By this training, it is possible that significant components, which determine the input-output relationships and realize the identity mapping of the input patterns, are acquired in the abstract layer.

After the initialization of the connection weights in the 3-layered network by this 5-layered network, the proposed system is trained by the back propagation method. In this training, errors in the FNN propagate backward into the 3-layered network. The connection weights in the 3-layered network are updated according to the errors from the FNN.

An advantage of the proposed system is that the pattern-symbol pairs in the fuzzy inference system work well for knowledge acquisition and for adaptation of the nonlinear projector. From the viewpoint of knowledge acquisition, it is desirable that relationships between symbols and fuzzy sets are as simple as possible. In the proposed system, the fuzzy inference system is designed to be simple with the abstract space being equally divided by fixed membership functions. The nonlinear projector is trained to adapt to the fuzzy inference system.

3 Simulation

The proposed system was applied to a control system for an autonomous robot with 5 distance sensors in 5 directions $(-90°, -45°, 0°, +45°, +90°)$. The robot infers a steering angle according to the inputs from the sensors, rotates, and then moves a constant distance D forward. The robot repeats this procedure again and again.

First of all, the robot was controlled using a fuzzy inference system designed by humans for acquisition of training data. The following is an example of the rules in this system:
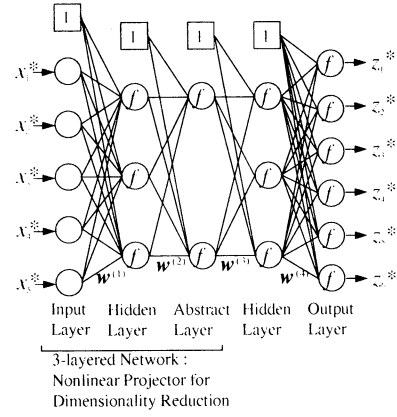
Figure 2: Initialization of Nonlinear Projector
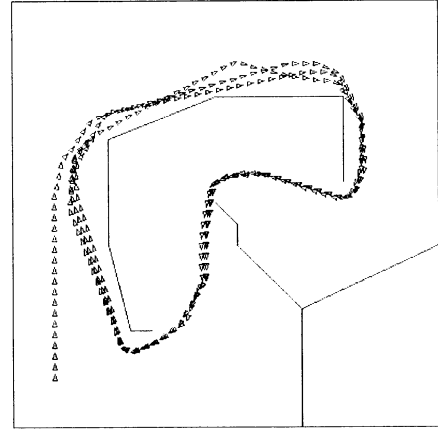
Figure 3: Robot's Trajectory for Training

- If x_1 =S, x_2 =M, x_3 =B, x_4 =B, x_5 =B then y =ZO.

This rule is hard to grasp intuitively because of 5-dimensional input.

Fig.3 shows the work space of the mobile robot and a trajectory using this fuzzy inference system. The proposed system was trained with 285 input and output patterns acquired from this trajectory.

The fuzzy inference system of the proposed system had three membership functions equally allocated on the universe of discourse of each abstract input. The proposed system was trained with this simply designed fuzzy inference system.

Fig.4 shows the abstract patterns in the abstract layer after the training of the 5-layered network with 285 training patterns of the trajectory in Fig.3. The solid lines were the centers of fuzzy boundaries given
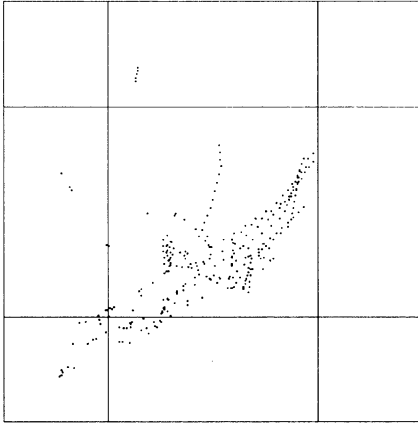
Figure 4: Initial Abstract Patterns

0.063 2	-0.0005	-0.0001
-0.9156	0.0021	0.0003
-1.0000	0.8434	0.1760
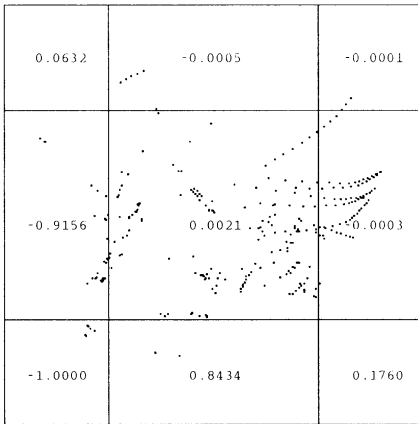
Figure 5: Acquired Abstract Patterns

by the membership function of the FNN.

Fig.5 shows the abstract patterns after the training of the FNN and the nonlinear projector. The mean squared error of the outputs was reduced from 0.059413 to 0.000189 by this training. The value in each of the subspaces shows the corresponding singleton in the consequent part. We can grasp the behavior of the system by observing abstract patterns in this abstract space.

Fig.6 shows the trajectory in the case where the robot was controlled 10,000 steps in the work space shown in Fig.3 by the proposed system. The proposed fuzzy inference system could work appropriately in spite of the simple design of the fuzzy inference system. This trained control system can not always work well in any work spaces, because the nonlinear projector was adapted to only the work space shown in Fig.3. If sufficient data are available beforehand, the
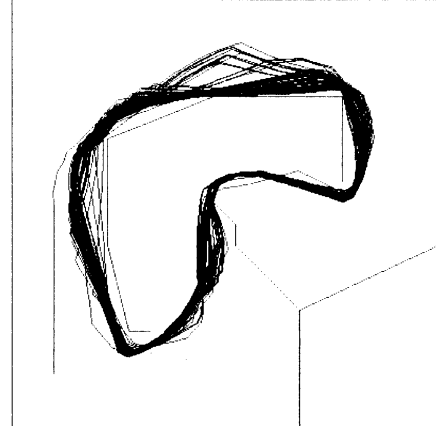
Figure 6: Trajectory Using the Proposed System

simple inference using the abstract patterns is more robust than the one using the original multi-dimensional patterns.

4   Conclusion

This paper proposed a fuzzy inference system with a nonlinear projector for dimensionality reduction to visualize relationships between the multi-dimensional patterns and the symbols. The proposed system was applied to a controller for an autonomous robot with multiple distance sensors. The robot's behavior was visualized on the 2-dimensional abstract space.

References

[1] N. Sakurai and T. Omori, "Image Recognition with Cooperation of Symbolic Inference and Neural Networks", Proc. of Int'l Conf. on Artificial Neural Networks'92 (1992)

[2] T. Ohmori, N. Yamanashi, "PATON: A Model of Concept Representation and Operation in Brain", Proc. of Int'l Conf. on Neural Network'94, pp.2227-2232 (1994)

[3] Y. Hattori, T. Furuhashi, "Retrieval of Patterns and Symbols Using Pattern-Symbol Network with Nonlinear Dynamics", Proc. of the 1997 Int'l Conf. on Neural Information Processing and Intelligent Information System, pp.408-411 (1997)

Performance of Information Recommendation Agent
Based on Reinforcement Learning in Society

Norihiko Ishitani, Tomoya Kinone and Shun'ichi Tano

Graduate School of Information Systems, University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan
Tel: +81-424-43-5601, Fax: +81-424-43-5681
E-mail: tano@is.uec.ac.jp

Abstract

We examine the problems with conventional approaches to find out a user's needed information from many different kinds of information. We analyze the problems and the weak points of the conventional methods to recommend a user's needed information, and propose a new method that supplements conventional methods. An evaluation of the new method is presented.

1. Introduction

The Internet, multimedia applications and digital devices have been ubiquitous in our society. Most people have their own PCs and can easily access to the World Wide Web site or on-line news service without being charged. Big changes are going to occur in broadcasting. From the technical point of view, it have been possible to broadcasting more than one thousand TV programs simultaneously by using digital satellite broadcasting systems.

Viewing the large number of programs is problematic, however, because the human capacity for information processing is limited. So, it is impossible to manage the large mass of programs. For example, the average viewer watches TV for only 3 hours a day [1]. Of cause if you do not have an interest in a program, you do not watch that program. Nevertheless, it is be difficult deciding which program one should watch. A system that can learn what information or programs a viewer wants or enjoys and that can then recommend them to the viewer would be a solution to the situations.

This paper examines problems with conventional information recommendation methods and proposes a new method and evaluates is.

2. Conventional Approaches and Their Problems

Conventional approaches [2] are content-based recommendation, collaborative recommendation and a combination of these two.

Content-based recommendation is a method whereby the system makes recommendations based on a user profile built up from programs the user watched in the past. The system is able to evaluate the user's response to the recommended programs and update the user's profile.

It can therefore adapt to the user's preference.

In collaborative recommendation, one's preferences are estimated according to their similarity to the profiles of friends who have the same or similar preferences. If some programs do not match in a user's profiles but do in a friend's profile, the system can recommend those programs.

It is impossible for the conventional recommendation methods to recommend information without the user's profile detailing what that user has enjoyed in the past. There are two problems with these approaches, as described in [3].

The first is that conventional approaches depend too much on the user's voluntary activity of accessing information. Although human's capacity of information accessing and processing is limited, amount of available information is drastically increasing. Users cannot access whole information. Therefore the conventional approaches have essential problems.

The second problem with the conventional collaborative recommendation is existence of certain information that cannot be recommended to user in spite of being in user's profiles. For example, in collaborative method, one's preferences are estimated according to their similarity to the profiles of friend who have the same or similar profiles. The information is gradually circulated by the collaborative recommendation scheme. Eventually the information spreads in the society. But in some cases, the sequence of the relation (i.e. such a sequence as person A is similar to B, and B is similar to C, C is similar to D, and so on) makes up the closed group. In this case, the information can not circulated beyond the closed group (see Fig. 1). This is the serious limitation of the collaborative recommendation. We named this boundary as "Collaborative Horizon".

We defined "Unknown Program" as those symbolic problems as follows.

Essential Problem "Unknown Program"

We classify two types of unknown programs that cannot be recommended by conventional methods.

(1) Complete "unknown program"

Conventional methods depend on possession of a

user's profile. So it would therefore be impossible to recommend completely new kinds of programs that nobody has seen in the past, since these programs would not exist in a user's profile.

(2) Collaborative horizontal "unknown program"

Even how actively system try to recommend programs, system cannot recommend some information that if it is not in one's or his group's user's profiles in spite of being in other group's profiles. This is caused by the horizon of the collaborative relation.
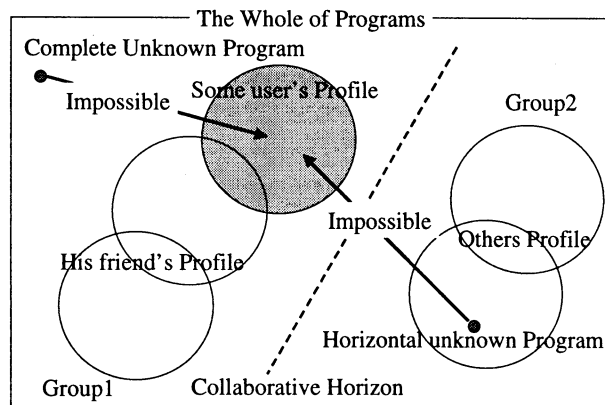
Fig.1 Existence of un-recommendable programs

3. Basic Concept of Our Model

To solve the problem of two unknown programs, we propose a recommending method based on reinforcement learning in society. This method incorporates the following three concepts [4].

(1) Trial Recommendation

In conventional methods, system cannot know that it should recommend unknown programs to a user or not. We propose that system *probabilistically tries* to recommend unknown program to the small size of the users, i.e. examinee, in order to get the user's response to the unknown programs.

(2) Collaborative Horizon Problem

When the collaborative horizons are formed, each horizon group should be treated as an independent society. Even though a program is not "completely new" in the whole society, the program may be "completely new" in one horizon group. Such a program should be *probabilistically* recommended to the examinee in the horizon group in the similar way explained above.

(3) Reinforcement Learning

Our system *probabilistically tries* to recommend unknown programs that it dose not know whether to recommend or not. It may be successful, or sometimes it annoys users. So we must clarify such the action principal as how often the system should recommend unknown program or how many users should be selected

as the examinees. If we set a good action principal, we will get good performance. The effectiveness of the action principal depends on two factors.

The first factor is the aspect of society. If the society includes many horizontal groups, we will get very good effect of trial recommendation. By contrast, if the society includes few horizontal groups, conventional methods are more effective than trial recommendation.

The second factor is trends in a user's preferences. If most users like finding their unspecified preference, we will get very good effect of trial recommendation too.

So it is very important for the system to be able to take account of the social aspects and trends of users, and then to be able to adjust the parameters of the action principal that is the most suitable for the current society.

An important parameter is the frequency at which system recommends unknown programs to the examinee. We named this parameter "trial recommendation rate (Tr)".

However the system is not given the characteristics the society and the users as a priori knowledge. So, the system has to adapt itself according to the experiences by the reinforcement learning. Through the trial-and-error experiences, the system gradually identifies social aspects and trends in the user's preferences. Our reinforcement learning method is based on the Q-learning.

4. Algorithm of Our Model

In this section, we describe the algorithm of our model. As explained in the previous section, our system consists of three types of agents.

The first agent recommends programs by the conventional methods and the programs that other agents ask to recommend.

The second agent identifies the horizontal groups and the collaborative horizon. It searches for two kinds of unknown programs. It asks the first agent to recommend the two types of unknown programs.

The third agent decides the parameters of the action principal by the reinforcement learning. In other words, this agent controls the system. For example, if the agent observes the examinee's strong dissatisfaction, it may reduce the trial recommendation rate (Tr).

The flow of the system is summarized below.
(Step 1) Initialize system
(Step 2) Get program data
    The contents of the TV program is represented by the word vector.
(Step 3) Recognize horizontal groups
(Step 4) Calculate unknown programs

Two types of unknown programs are identified.

(Step 5) Select examinee from target users

　　If there is unknown program, select the examinee from the whole society or the horizon group according to the action principal.

(Step 6) Recommend by conventional method

(Step 7) Recommend by trial method

　　If there is unknown program, the trial recommendation is evoked according to the action principal.

(Step 8) Update users' preference

　　Agent evaluates the users' response and updates the users' preference.

(Step 9) Update system parameters

　　Agent modifies the parameter of the action principal according to result by Q-learning.

(Step 10) Go to *Step 2*

5. Simulation and Evaluations

To evaluate the feasibility and effectiveness of our method, we tested it by using a simulated recommending system.

In this system, we use the word vectors to represent programs such as TV programs. We prepare two sets of word vectors. One set represents the user's true preference. The other represents the user's estimated profile.

5.1 Evaluation method

We use the following index, successful recommendation index H, to evaluate the system.

$$ H = \frac{\left(\sum wa_i \cdot a_i + \sum wb_i \cdot b_i + wc \cdot c + \sum wd_i \cdot d_i\right)}{\sum a_i + \sum b_i + c + \sum d_i} $$

The a_i, b_i, c, and d_i stand for the count of the TV-program. The meanings are listed below.

　　a_i: total number of good recommendation

　　　　a_0: by the content-based recommendation

　　　　a_1: by the collaborative recommendation

　　　　a_2: by the trial recommendation (ours)

　　b_i: total number of wrong recommendation

　　　　b_0: by the content-based recommendation

　　　　b_1: by the collaborative recommendation

　　　　b_2: by the trial recommendation (ours)

　　c: total number of neither good or wrong

　　d_i: total number of un-recommendation

　　　　d_0: if recommended, it is good.

　　　　d_1: if recommended, it is wrong.

The wa_i, wb_i, wc, and wd_i show the importance of the counts of a_i, b_i, c, and d_i respectively. For example,

$\{wa_0, wa_1, wa_2, wb_0, wb_1, wb_0, wc, wd_0, wd_1\} = \{1, 1, 1, -2, -2, -2, -0.1, 0.5, -0.5\}$ represents the characteristics of the society. In this society, wb_i is double of wa_i. This means that the penalty of the wrong recommendation is heavy.

5.2 Comparing the conventional method and our method

According to the total number of recommendation and successful recommendation and the rate of successful recommendation, we compared with Content-Based method (1), Content-Based + Collaborative method (2) and Content-Based + Collaborative + Trial (without Q-learning) method (3) (Table.1).

Table.1

	success	total number	rate
(1)	5126	7115	0.72
(2)	5966	9636	0.62
(3)	7234	12020	0.60

In comparison with conventional method (1)(2) and our method (3), the increase in the total number of recommendation can be regarded as the high activity of the recommendation. The increase the number of successful recommendation means that our method can recommend the hidden TV-program. In other words, our method revealed the users' hidden preference and embedded it into the user profiles.

On the other hand, the rate is down. We thought this is because the parameters of the action principal were inappropriate. We must find out more suitable parameters. But note that the decrease in the rate is relatively small to compare to (2).

5.3 Effect of trial recommendation rate

We gathered the data on the successful recommendation index H in various situations. We found that the trial recommendation rate strongly affected the rate of successful recommendation index H. Although the relation between them is nonlinear, the best unique Tr exists for each environment (Fig. 2).

Once the graphs are given, it is easy to find out the best Tr. But usually they are not given. So we must make the behavior of the system change according to the current environment. Our system realizes this mechanism by the reinforcement learning.

5.4 A system performance against the change in the environment

To evaluate the feasibility of the reinforcement learning method, we varied the testing environment (i.e., characteristics of the society) while the experiment was
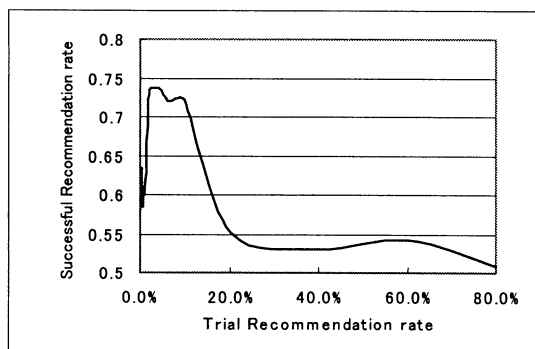
(a) condition 1
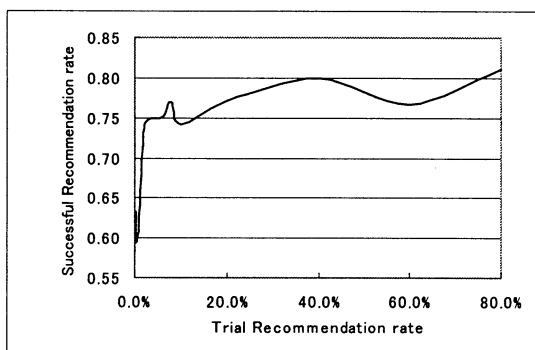
(b) condition 2
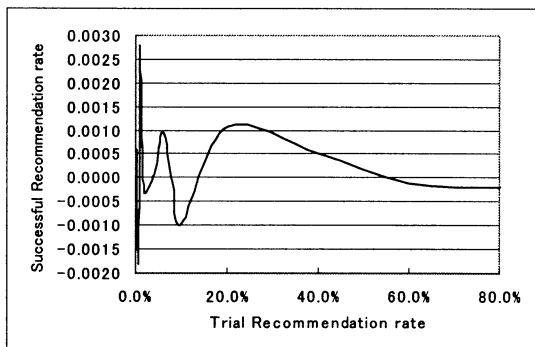
(c) condition 3

Fig 2 Three kinds of Conditions

still in progress.

We found that in a system with the reinforcement leaning method, when the environment changed, the system could adapt itself to the new environment. But in a system that did not use the reinforcement learning method, if could not adapt itself to the new environment (Fig. 3). We confirmed the effectiveness of reinforcement learning method in our method.

6. Future Works

Several problems still remain. Typical problems are:

(1) Suitable attributes and actions of Q-table

We adopted a reinforcement learning method. But current Q-table is not perfect. For example, it is not sure
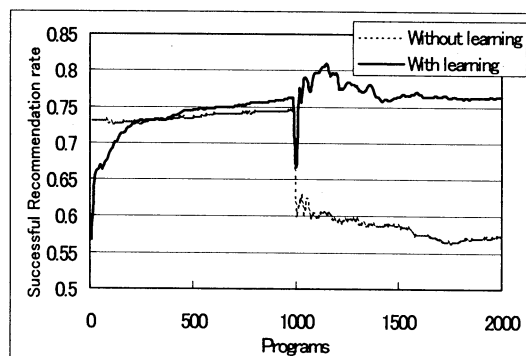
Fig 3 environment change

that the variety of the attributes is sufficient to decide the suitable system actions.

(2) How to select the examinee

The system did not recommend unknown programs to all users but the selected users. So it is more preferable to select the examinee based on the user's personality.

(3) How to represent of society and users

We took account of the characteristics of the society and the individuals. But the model is too simple to cope with the real world environment.

7. Conclusion

Problems with conventional approaches were made clear by analyzing conventional methods of recommendation. Introducing reinforcement learning in society enabled the system to recommend two kinds of unknown programs.

As listed in future works, however, there are many problems that remain to be solved, further research is needed.

References

[1] T. Nagaya: Information and Brainware: A Theory of the Human Capacity for Information Processing and the Basic Principle of "Preference-selection" Behavior, The Annual Bulletin of Broadcasting Culture Research, vol. 40, 1995, pp. 1-102.

[2] M. Balabanovic and Y. Shoham: Content-Based, Collaborative Recommendation, Communication of The ACM, Vol. 40, No. 3, 1997, pp. 66-72.

[3] T. Kinone, N. Ishitani and S. Tano: Proposal of TV Program Recommendation based on Reinforcement Learning in Regional Society and Initial Evaluation, Fuzzy System Symposium, 1999, pp. 767-770, (in Japanese).

[4] N. Ishitani, T. Kinone, and S. Tano: Analysis of Propagation Process in Recommendation Method Based on Reinforcement Learning in Regional Society, Electronics, Information and Systems Conference, 1999, pp. 329-330, (in Japanese).

Agent Systems Driven by Intentional Reasoning

Tomohiro Takagi*, Masao Mukaidono* and Toru Yamaguchi**

* Dept. of Computer Science, Meiji Univ.
  Higashi-Mita, Tama-ku, Kawasaki-shi, Kanagawa 214-8571, Japan
  takagi@cs.meiji.a.jp, masao@cs.meiji.ac.jp
** Dept. of Information Science, Utsunomiya Univ.
  2753 Ishii-cho, Utsunomiya-shi, Tochigi 321-8585, Japan
  yamachan@sophy.infor.utsunomiya-u.ac.jp

Abstract

We describe a method of conceptual matching for retrieving information that meets a user's intentions. We also propose a method for representing common concepts by fusing conceptual fuzzy sets (CFSs) with ontology. The matching is done based on the region of the ontological structure covered by the CFS representing the user's intention as well as on the original keywords.

We apply the proposed method to an agent which recommends TV programs to watch. The recommended programs have EPGs (Electronic Program Guide) similar to those of previously watched programs or that contain words matching the learning data. Practical examples demonstrate that the proposed system can recommend TV programs matching the user's tastes.

Keywords; fuzzy, intention, agent, ontology

1. Introduction

Information retrieval is generally done by using keyword matching, which requires that for words to match, they must be the same or synonyms. But in an actual search, a user will often input keywords that are only examples, and such keywords may not represent the user's intentions necessarily and sufficiently. Therefore, not only the information that matches the keywords exactly, but also information that is related in meaning to the user's intentions should be retrieved. For example, in selecting the place for a vacation, even if the user indicates "Hawaii and Tahiti", his or her intention is likely "a tropical resort" and not simply Hawaii and Tahiti. Therefore, other tropical resorts, such as Fiji, should be added to the candidates for selection.

This problem cannot solved by simply using synonyms. Although many approaches to identifying a user's true intentions have been proposed[1-5], they have limited flexibility in determining correspondence because of their previously constructed procedures. To overcome this problem, we previously proposed conceptual fuzzy sets (CFSs)[6,7].

In this paper, we describe the application of CFSs to intentional reasoning and conceptual matching. In Sec. 2,

we introduce CFSs. In section 3, we describe the fusion of CFSs and ontology to achieve intentional reasoning and conceptual matching. In Sec. 4, we describe an agent that suggests TV programs a user might enjoy. It recommends programs that are similar to those previously watched by the user. We summarize our paper in Sec. 5.

2. Meaning Representation using CFSs

A CFS is realized as an associative memory in which a node represents a concept and a link represents the strength of the relation between two (connected) concepts. The activation values agreeing with the grades of membership are determined through this associative memory. In a CFS, the meaning of a concept is represented by the distribution of the activation values of the other nodes according to the theory of "meaning representation from use" proposed by Wittgenstein[11]. The distribution evolves from the activation of the node representing the concept of interest. The image of a CFS is shown in Fig. 1.
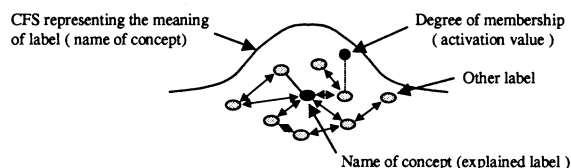
Fig. 1 A conceptual fuzzy set represented by associative memories

The activation of nodes produces a reverberation, and the system energy is stabilized at a local minimum where corresponding concepts are recollected as a result. This is done using a weight matrix encoded from stimulus-response paired data. In this paper we use bidirectional associative memories (BAMs)[9,10] because of the clarity of the constraints used for their utilization. The BAMs encode the CFS by storing corresponding pairs of elements at each layer in terms of a synaptic weight matrix.

Example: CFS representing "fat"

Simple examples of CFSs are shown in Fig. 2 A BAM consists of two layers: the upper layer includes concept nodes "fat" and "tall", and the lower layer

includes instance nodes, such as "Dan" and "George". The activation of node "fat" creates a distribution of activations of the nodes in the lower layer. In this example, fuzzy set "fat" is given as follows: fat = {0.4/Dan, 0.9/George, 0.3/Mark, 0.0/Paul} The ratio of the black portion in each node represents the grade of activation in region [0,1] which corresponds to the degree of membership in "fat". The black arrows indicate promoting links, and the gray arrows indicate degrading links. Each node has a sigmoid function.
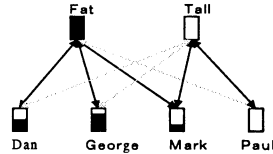
Fig. 2 The Conceptual Fuzzy Set "fat"

3. Intentional Reasoning and Conceptual Matching

An ontology[8] is an organization consisting of words describing common ideas. By fusing CFSs with ontology, a region of words commonly relevant to the original keyword is formed. A search for information referring to a region of the CFS derived from keywords activates information retrieval that satisfies the user's actual intention, whereas an ordinary search with superficial keywords may have problems. The information retrieved shows the same degree of relevance as the words in the CFS.

3.1. Approximate Reasoning using CFSs and Ontology

As we see above, the meaning of the concept is translated into the expression indicated by the distribution of activation in each layer, and the following reasoning can be realized in CFS (Fig. 3):
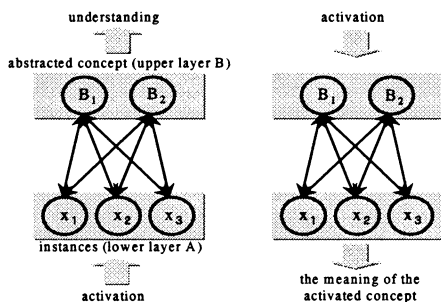
Fig. 3 Meaning representation in multiple layers

1) The activation of a lower layer determines the activations of an upper concept and it corresponds to recognition or understanding. This property works as a intentional reasoning from user's fragmentary signs (ex: keywords for information retrieval).
2) In contrast, propagation which arise from the activation of an abstracted concept show its meaning in the concrete layer. This property works as recommending examples which suit to the user's intention.

Further, due to its bidirectional features, CFS can realize the parallel processing to support the fusion of bottom-up and top-down processing. At that time the context described in the upper layer depresses the meaningless patterns of distribution of activation and promotes the meaningful patterns, satisfying the context, of activations in lower layer. This context sensitive processing provides us with an accurate intentional reasoning. It uses the context to eliminate vagueness which may come from noisy and vague data and which could otherwise cause misunderstandings.

3.2. Conceptual Matching using CFSs and Ontology

3.2.1. Recollection of Concrete Examples from an Abstract Concept

In this conceptual matching using abstract concepts, the elements of the CFS that are recalled by the associative memory are also used for matching based on their degree of membership.

3.2.2. Recollection of Other Similar Subjects from Instances

In conceptual matching that starts with one or more elements of the CFS, superordinate concepts are first recalled from instances and they recall other elements in the CFS. These recalled elements are also used as keywords for information retrieval. This activation of elements activates related abstract concepts, which in turn activates other elements.

3.2.3. Recollection with Multiplex View Points

If the original keywords from the user are "Hawaii" and "Palau", the user's intention may actually be "near Japan" and also "tropical resort." In contrast, if the keywords are "Singapore" and "China," the intention may be "near Japan" and "sightseeing place." In the former case, the CFS's region of the user's intention is generated by the overlap of criteria "tropical resort" and "near Japan," and the CFS suggests another candidates, such as Fiji. In the latter case, the intention is generated by the overlap of "sightseeing place" and "near Japan", so Korea is suggested as another candidate. Table 1 shows the change in the CFS depending on the user's intention based on the input examples above, where the underlined values are the ones with greater relevance.

Table 1 Change in matching region

	tropical resort near Japan	sightseeing place near Japan
Maldives	0.01	0.02
Singapore	0.09	0.16
China	0.11	0.19
Korea	0.12	0.20
Hawaii	0.17	0.14
Palau	0.17	0.12
Fiji	0.16	0.12
Caribbean	0.09	0.06

4. Agent for TV Program Selection

We applied the proposed matching method to an agent for recommending TV programs that match a user's intention (user's tastes). With the hundreds of TV channels now available, it can be difficult to find programs of interest. Our approach to this problem is to use an agent to identify suitable TV programs by using the EPG (Electronic Program Guide: information about the program) which is also broadcast. Here, for simplicity, we only consider drama-type programs.

4.1. Description
4.1.1. System Outline

The agent learns the user's tastes from the contents of the TV programs previously watched and recommends programs matching these tastes. Actually, a CFS agreeing with user's tastes is inductively constructed on the ontological structure from the EPGs of the programs watched in the past. The agent then recommends programs having words in their EPGs with a higher membership grade in the CFS (see Fig. 4). EPGs containing title, date, time, actor/actress, story, etc. are used by agent to recommend programs.
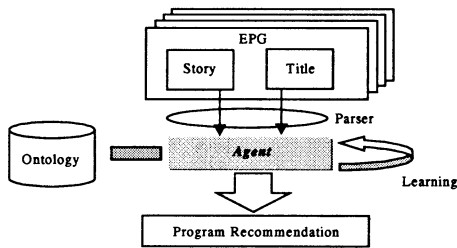
Fig. 4   System outline

We used an ontological structure with three layers. The top layer contains 11 words representing "category." The second layer contains 3,670 words describing the categories in the first layer. These layers are connected by a BAM[9,10]. The third layer contains synonyms of the words in the second layer.

4.1.2. Learning User's Intention (taste)

A user's taste are learned from the words in the EPGs of previously watched programs. The BAM between two layers was constructed using Hebbian learning. For example, when a user watched a program whose EPG included the words "trouble", "evil," "police station," and "detective," the connections between these four words and "crime" as one of categories are reinforced.

4.1.3. Recommending Programs

The learned CFS agrees with the user's tastes, it is derived by activation of the "user's tastes" node after learning. The degrees to which programs match the tastes are the averages of the activation values of the words included in the EPGs of watched programs.

The example shown in Fig.5 shows how two programs are matched to a use's tastes. The values attached to the words in the category and EPG layers represent the degrees to which they match the user's tastes. The values for the two programs show the degrees to which they match the use's tastes. This example shows how the proposed approach can recommend programs whose EPGs does not include the same words used in learning process, but include words related in a general sense, i.e., programs that the user should find interesting.
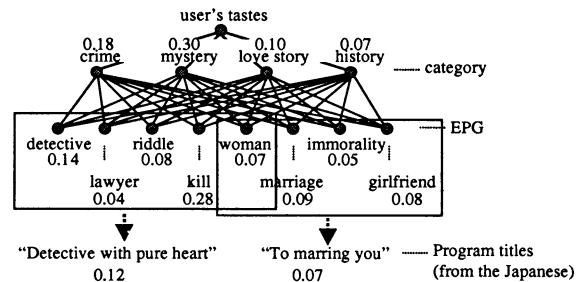
Fig. 5   Program   recommendation according to matching degrees

4.2. Practical Examples

We simulated learning and program recommendation by using 40 candidate programs in various categories.

4.2.1. Recommendation Using One Factor

The top recommendation in each category after learning is shown in Table 2. The values before learning were derived only from the initial connections in the ontology.

Table 2   Recommendations before and after learning

category	program name	before learning	after learning
crime	"Aristocratic detective"	0.039	0.094
mystery	"Cat mystery"	0.053	0.064
love story	"Heike romances"	0.031	0.010
history	"To marring you"	0.059	0.014
period drama	"Rough shogun"	0.034	0.035

After learning, the "crime" program had a significantly larger value than the others, whereas before learning all programs had almost the same value. The "mystery" program also had a large value to some extent because these two categories have a similar atmosphere.

The top ten recommended programs are shown in Table 3. They are mainly crime and mystery programs. The one history program "Rough shogun" is listed because its EPG includes the words "kill", "culprit," and "smuggling," which are related to crime. This shows that the agent can match programs according to content as well as category.

Table 3　Top ten recommendations based on one factor

	program name (category)	matching degree
1.	"Spa murder case" (crime)	0.10
2.	"New town police station" (crime)	0.09
3.	"Aristocratic detective" (crime)	0.09
4.	"Cat mystery" (mystery)	0.06
5.	"Detective with pure heart" (crime)	0.05
6.	"Rough shogun" (history)	0.04
7.	"Detective Konan" (mystery)	0.03
8.	"Women refusing marriage" (crime)	0.03
9.	"Taxi driver's diary" (mystery)	0.03
10.	"Geisha sisters" (crime)	0.03

4.2.2.　Selection Using Several Factors

In Sec. 3 we explained how a CFS indicate the region of intention based on several criteria by the overlap of activations. Table 4 shows the result of using the criteria "emotion" and "category" against the same 40 programs. We added ten words representing emotion to the top layer. After relearning the user's tastes not only "crime" but also "human warmth" programs were highly recommended. This shows that the agent can consider several criteria and apply them to actual applications.

Table 4　Top ten recommendation based two factors

	program name (category)	degree of "human warmth"
1.	"New town police station" (crime)	0.22
2.	"Aristocratic detective" (crime)	0.24
3.	"Stakeout" (crime)	0.38
4.	"Detective with pure heart" (crime)	0.41
5.	"Bare detective" (crime)	0.22
6.	"Murder railroad" (crime)	0.00
7.	"Women refusing marriage" (crime)	0.28
8.	"Cat mystery" (mystery)	0.00
9.	"King's restaurant" (comedy)	0.01
10.	"Murder case in" (crime)	0.04

5.　CONCLUSION

We have described a method of conceptual matching for retrieving information that meets a user's intentions. We also described a method for representing common concepts by fusing conceptual fuzzy sets (CFSs) with ontology. The matching is done based on the region of the ontological structure covered by the CFS representing the user's intention as well as on the original keywords.

We applied the proposed intention reasoning method to an agent which recommends TV programs to watch. The recommended programs have EPGs similar to those of previously watched programs or that contain words matching the learning data. Practical examples demonstrated that the proposed system can recommend TV programs matching the user's tastes.

References

[1] J.L. Robert, A. Sheila, H.M. Robert, and W. Keith (1995), "Living Room Culture – An Authropological study of Television Usage Behaviours," Proceeding of the Human Factors and Ergonomics Society 39th Annual Meeting, 326-330.

[2] D.J. Litman and J.F. Allen (1987), "A Plan Recognition Model for Subdialogues In Conversations," Cognitive Science, Vol. 11, 163-200.

[3] Y. Yamashita, H. Yoshida, T. Hiramatsu (1993), Y. Nomura and R. Mizoguchi, "MASCOTS II: A Dialog Manager In General Interface for Speech Input and Output," IEICE Transactions on Information and Systems, Vol. E76-D, No. 1, 74-83.

[4] R. Kass and T. Finin (1988), "The Need for User Models In Generating Expert System Explanations," International Journal of Expert Systems, Vol. 1, No. 4, 345-375.

[5] T. Finin (1988),"Default Reasoning and Stereotypes In User Modeling ," International Journal of Expert Systems, Vol. 1, No. 2, 131-158.

[6] T. Takagi, A. Imura, H. Ushida, and T. Yamaguchi (1995), "Conceptual Fuzzy Sets as a Meaning Representation and their Inductive Construction," International Journal of Intelligent Systems, Vol. 10, 929-945.

[7] T. Takagi, A. Imura, H. Ushida, and T. Yamaguchi (1996), "Multilayered Reasoning by Means of Conceptual Fuzzy Sets," International Journal of Intelligent Systems, Vol. 11, 97-111.

[8] T.R. Gruber (1992), "Ontolingua: A mechanism to support portable ontologies," Technical Report KSL Stanford University, Knowledge Systems Laboratory, 91-656.

[9] B. Kosko (1987), "Adaptive Bi-directional Associative Memories," Applied Optics, Vol.26, No.23, 4947-4960.

[10] B. Kosko (1992), "Neural Network and Fuzzy Systems," Prentice Hall.

[11] Wittgenstein (1953), "Philosophical Investigations," Basil Blackwell, Oxford.

[12] "The Full Text Retrieval of TV Program," http://qzc.co.jp/DRAMA/

Generalization ability of Minimum Description Length for Multiple-Organisms Learning by Genetic Programming

Yoshida Akira(akira-yo@is.aist-nara.ac.jp), Graduate School of Information Science,
Nara Institute of Science and Technology ; 8916-5 Takayama Ikoma Nara 630-0101 Japan

Abstract

Exhaustion of exploration ability caused by an excess of the size of tree by genetic programming could be seen on some cases. Minimum Description Length principle can not only decrease the size of the resulting evolved algorithm or tree, but also increases their generality and the effectiveness of the evolution process. We give an example of generalization ability by means of minimum description length with mutiple-organisms learning by genetic programming.

1 Introduction

Genetic programming [1] is an interesting extension of Genetic Algorithms. In genetic programming, a population of programs, that are represented by trees in a task specific language, evolve to accomplish the desired task. One of the advantages of genetic programming over genetic algorithms is noted as its dynamic tree representation. The interpretation of a particular expression tree is performed without consideration to the position of the allele in genetic programming. On the contrary, in genetic algorithm, the typical interpretation function uses a positional encoding where the semantics of the interpretation are tied to the position of the bit in the genotype.

We have been studying adaptive organisms in evolving populations, and finding relations between learning and evolution by genetic programming[2]. This is next step for us to use genetic programming more effectively by means of minimum description length principle. We would like to extend our study by means of genetic programming in order to process representation of knowledge, such as program and tree.

First, we explain methods of simulation using genetic programming to observe evolution and learning of collective behavior of organisms. Next, we discuss the results of the simulation of co-operation, evolution and learning comparing the results of simulations. Finally, an example that exhaustion of exploration ability caused by an excess of the size of tree is corrected by minimum description length principle is explained.

2 Genetic Programming for Learning by Organisms

Genetic programming uses three operators that change the structure of a tree, mutation, inversion, and crossover. These are natural extension from Genetic Algorithms operators that deal with only the bit strings. The reproduction phase of genetic programming is also nearly same as the reproduction system of Genetic Alrotithms.

Here we describe an approach using co-evolving organisms in cooperation to generate the emergence of job separation among the organisms. Genetic programming generates a program which specifies the behaviour of organisms as the form of tree, and evolves the shape of the tree. We have three nonterminal symbols(functions) and three terminal symbols(actions) in order to make nodes of a tree. This tree controls movements of an organism.

Three kinds of functions are defined as follows:
(a) If organism sees a food in front of him then perform first argument, else perform second argument[IfFoodAhead],
(b) Perform two arguments in order[Prog2],
(c) Perform three arguments in order[Prog3].
Three kinds of action are defined as follows:
(a) Turn right at an angle of 90 degrees,
(b) Turn left at an angle of 90 degrees,
(c) Move forward a box per unit time.

An organism moves according to the structure of a tree that consists of above three nonterminal symbols and terminal symbols. An example of a simple tree is shown below table 1.

We extended "sgpc1.1" and the Ant system written by Iba Hitoshi[3] to simulate behaviour of multiple organisms.

2.1 Feeding Simulation for Organisms

Our experimental domain consist of a square divided into 32 x 32 boxes. On the first experiment the nest

site of organisms is at the upper left and foods are scattered in the domain on a trail. Organisms can search for their foods as long as their energy last. An organism can change his moving direction to east, west, north, and south, but can not change the direction diagonally. He moves forward a box a unit time, and can look for a food in front of him with his sensor. The organism spends a unit of energy whenever he takes an action: move forward, right, or left. Organisms must search foods efficiently because their energy is limited and they are killed when they spend all energy. Since fitness is evaluated as the number of foods organisms could not find, the smaller value of fitness means the better feeding by organisms. Here, organisms are like any social insect that live in a group, for example, ants, bees, or a kind of cockroaches.

Each organism starts his action in turn after an unit movement of previous organism has completed. The start point of all organisms is the nest site, and the initial moving direction of each organism are different from previous organism at an angle of 90 degrees. This experimental domain is not a torus and the east end is not continued to the west end, nor the north end continue to south end. The organism which has reached to the east end must turn to other directions. Figure 1(right) shows the trails made by a simulation with four organisms at 22nd generations with the value of fitness 36. There are trajectories of four organisms, where the newer trajectory(bigger number) overwrote the older trajectories partially. We could see that the more number of organisms the much area is searched. Here we used the "Santa Fe trail" for the initial foods pattern with upper left nest site.

2.2 Co-operation and Job separation

We could see two kinds of emergence of cooperative behaviour as the number of organisms increases : improvement of feeding efficiency and job separation. On the following experiments we use many food patterns where the foods are scattered in a nearly round shape or isolated island shape. Figure 1 (left and center) shows two kinds of initial food patterns and the place of nest site for simulations of emergence of the job separation. We tried two cases : that the nest site is set on left upper corner in the domain and at the center of foods.

(A) Improvement and saturation of feeding efficiency by co-operation

Figure 2(left) compares the fitness value in each generation with various number of organisms. The values of fitness are the lower the better. We can see if the number of organisms were eight or more the

fitness value converged to less than 20 after nearly 200th generation. On the contrary, the fitness values of one-organism and 4-organisms never converge and stay near the initial fitness values even at the 200-th generation. Although one or four organisms could not evolve to raise fitness in this environment eight or more organisms could evolve by co-operative behaviour. But the important point here is that more than 16 organisms could not raise the efficiency of their work as a whole no matter how they devised plans. If there were too many organisms some extra organisms had no choice but to be out of work and ramble without finding foods. This is the reason why extra organisms must invent a new job and may be the origin of the job separation.

(B) Division of worker organism and guard organism

Figure 2(right) shows an example of the emergence of division between worker organisms and guard organisms. The nest site is set on left upper corner(Figure 1, left). The trajectories from first to 13th organism are deleted here to make it easy to understand. The 16th organism traveled the world diagonally to find foods until generation=24, but he began to ramble around the nest site after generation=85. On the contrary, the 14th organism rambled around the nest site until generation=24, but evolved to travel around the world diagonally after generation=85. They replaced their jobs after about generation=50.

3 Minimum Description Length principle rescues from Overfitting

Kinnear proposed to add the size of trees to the fitness measure[4]. This method not only decreases the size of the resulting evolved algorithms or tree, but also increases their generality and the effectiveness of the evolution process. Minimum description length principle may be defined as : the sum of the description length of error and the description length of tree should be minimum. The new fitness F-mdl that include minimum description length principle is computed as follows :

F-mdl[i] = fitness[i] + (number-of-nodes[i] × parsimony-factor),

where, "number-of-nodes[i]" means the number of nodes in i-th tree, and "parsimony-factor" is a positive constant. fitness[i] is a number of foods left over as stated previously. The product of "number-of-nodes" and "parsimony-factor" should be smaller since fitness

is the smaller the better. This fitness definition can be said a tradeoff between the complexity of model and the error range.

If the parameters were not appropriate on our simulations by GP, we could see some situations that tree(program) grow tremendously, on the other hand fitness never improve. Figure 3 shows the fitness values vs generation both on "parsimony-factor=0.0" (left) and "parsimony-factor=0.5"(right). Here we have two nest sites and two clusters of foods in the experimental domain(fig 1,middle). As the number of organisms increases, the convergence value does not get smaller on figure 3(left). This result seems clearly erroneous since fitness should be proportional to number of organisms as we have confirmed on many experiments[2]. On the contrary, we can see figure 3(right) with the value of "parsimony-factor=0.5" shows the normal result; fitness generally decreases as the number of organisms increases. The fitness values of 8 or 32 organisms are converged to lower values, which seems to be a right result, however the fitness values of 2 or 4 organisms show chaotic vibration.

Table 1 shows program(tree) sizes and fitness values on 100-th generation with three kinds of numbers of organisms and values of parsimony-factor. Here the program(tree) size means the total number of functions and actions in a tree. We can see, on the case of 8-organisms, the program size was decreased from 480 steps (parsimony-factor=0.0 ; abnormally big tree) to 19 steps(parsimony-factor=0.5 ; small tree) on 100-th generation. This small program is shown under Table 1. On this experiment all the fitness values have chaotic vibration when the value of parsimony-factor is greater than 1.0. We can see that the program size decreases abruptly as the value of parsimony-factor increases.

This is one of examples that exhaustion of exploration ability caused by an excess of the size of tree is corrected by minimum description length principle.

4   Results and Conclusion

(1) Efficiency of finding foods gets better as the the number of organisms increases until a limit value (emergence of co-operation).
(2) When the number of organisms reaches the upper limit extra organisms begin another jobs except finding foods (emergence of the job separation).
(3) Exhaustion of exploration ability caused by an excess of the size of tree by Genetic Programming is corrected by Minimum Description Length principle

We have explained the emergence of co-operation, job separation, and division of territories in a simple world controlled by genetic programming. Saturation of efficiency when the upper limit of the number of organisms is reached to co-operate the work could be the cause of job separation and division of territories.

This paper shows that some situations that tree(program) grow tremendously, but fitness never improve were rescued from overfitting by minimum description length principle. This is a simple and powerful method for improvement of algorithms of genetic programming.

Acknowledgment

References

[1] Koza JR (1992) Genetic Programming, On the Programming of Computers by means of Natural Selection,MIT Press
[2] Yoshida A (1999) Multiple-Organisms Learning and Evolution by Genetic Programming, AJWIES'99
[3] Iba H (1996)Genetic Programming, Tokyo Electric University Press(Japanese)
[4] Kinnear EK(1993) Generality and Difficulty in Genetic Programming, ICGA'93. Proc. p.287

Table 1. Program(Tree) sizes and Fitness values on 100-th generation

		parsimony-factor=0.0		parsimony-factor=0.5		parsimony-factor=1.0	
Number of organisms		Program size	Fitness	Program size	Fitness	Program size	Fitness
2-organisms		446	4	35	46	6	62
4-organisms		465	17	48	15	5	67
8-organisms		480	16	19	6	31	26

(PROG2 (PROG2 FORWARD (PROG2 (IfFoodAhead FORWARD RIGHT) (PROG2 FORWARD (IfFoodAhead FORWARD (PROG3 RIGHT (PROG3 FORWARD FORWARD FORWARD) LEFT))))) FORWARD)
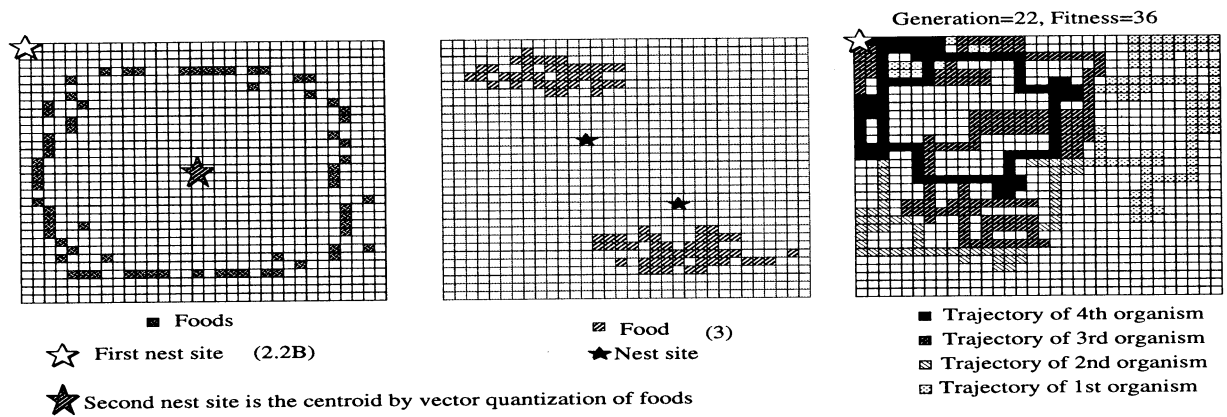
Figure 1: Two food patterns and nest site(left,middle); Trajectories of four organism with upper left nest(right).
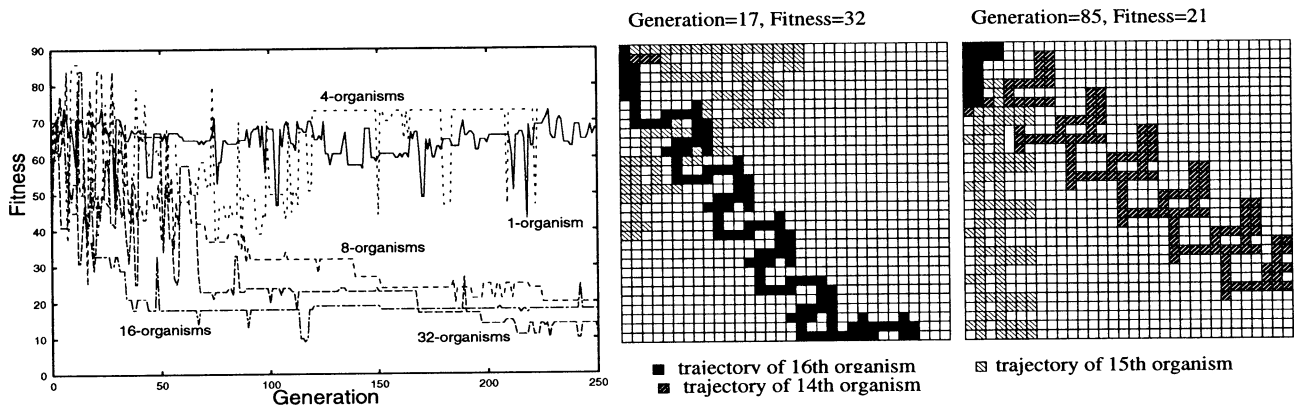
Figure 2: Fitness value vs. Generation; if the number of organisms were more than or equal to 8 the fitness value converged to less than 20 after nearly 200th generation(Santa Fe trail ;left). Trajectories of 16-organisms with upper left nest; Emergence of worker and guard(right two).
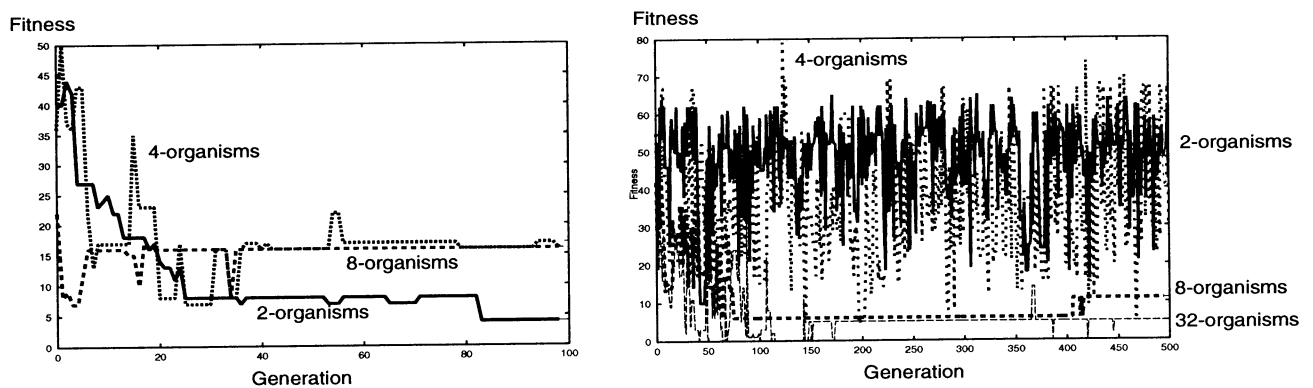
Figure 3: The result by Minimum Description Length principle : parsimony-factor=0.0 (left),=0.5(right)

Emergent Design of Optimal Postures
for Mobile Manipulators
by using A Java Interactive GA Viewer

Kiyoharu Tagawa*, Jang-Myung Lee†,

Hiromasa Haneda* and Man-Hyung Lee‡

*Dept. of Electrical and Electronics Engineering, Kobe University
Kobe 675-8501, Japan
†Dept. of Electronics Engineering, Pusan National University
Pusan 609-735, Korea
‡School of Mechanical Engineering, Pusan National University
Pusan 609-735, Korea

Abstract

A man-machine cooperative design tool, Interactive GA Viewer (IGAV), is presented. The purpose of developing IGAV is to realize the emergent approach for design of artificial systems. IGAV offers a graphical user interface to use GA interactively. Especially, IGAV has a unique function to deal with constrained optimization problems effectively. As a practical application, IGAV is used to design the optimal postures of mobile manipulators. Mobile manipulators have kinematic redundancy created by the addition of the degrees of freedom of the mobile platform and those of the manipulator. Therefore, designing the optimal positions and configurations of them is very difficult problem. Experimental results show an advantage of using IGAV for the complex optimization problem.

1 Introduction

Mobile manipulators, i.e., manipulators mounted on mobile platforms, are attracting significant interest in the industrial, medical, military, and public service communities, because their large-scale mobility combined with manipulation abilities provide capabilities and efficiency in material-handling tasks[1]. An important characteristic of mobile manipulators is their particular kinematic redundancy created by the addition of the degrees of freedom of the mobile platform to those of the manipulator. Therefore, they have to be optimally positioned and configured for maximum performance. Even though several performance measures advocated for conventional fixed-base manipulators can be used to evaluate the configurations of mobile manipulators[2], it is more difficult to design the optimal positions and configurations of them.

In order to cope with such a difficult problem of artificial system design, a methodology called "emergent design" has been proposed[3]. In the scheme of emergent design, the design procedure is defined as an inverse problem conceptually. Then, the inverse problem is solved by the iteration of forward mapping where human designers should play an important role. Therefore, not only evolutionary computation such as Genetic Algorithm (GA) but also man-machine cooperative interface is indispensable technique to realize the emergent design.

In this paper, we present a man-machine cooperative design tool based on GA, Interactive GA Viewer (IGAV), for realizing the emergent design. Then, we apply IGAV to the design of optimal postures for mobile manipulators that is formulated as a constrained optimization problem. There have been developed various type of GA simulators which use graphics for easy operation and plain display of simulation results[4, 5, 6]. IGAV has inherited some excellent functions from the conventional GA simulators. Furthermore, IGAV has the merit of solving constrained optimization problems more effectively. Penalty functions are the most common technique used in GAs for constrained optimization problems[7]. However, there is no general guideline on designing penalty function, and constructing an efficient penalty function is quite problem-dependent. The characteristic of IGAV is it unique function to modify such a penalty function interactively according to the behavior of GA.

2 Interactive GA Viewer

The developed IGAV is design by Object Modeling Technique (OMT)[8] and written in Java language. One of the advantages of using Java language to implement IGAV is its platform independence. In other words, IGAV works in the same way on any computer that has a Java interpreter. Figure 1 shows the data flow diagram of IGAV, i.e., the functional model used by OMT, in which an ellipse denotes process and a rectangle denotes data-store. As you can see in Fig.1, IGVA consists of two main processes, namely GUI (Graphical User Interface) and GA, which are invoked as Java-Threads respectively and run in parallel.

The "GA-process" mainly executes a numerical optimization GA program based on the floating point coding[9]. At the present time, three crossover operations, namely one-point, two-point and uniform crossovers, are provided. As we shall see later, we can change not only the GA parameters, namely crossover-rate, mutation-rate and population size, but also the type of genetic operations interactively.

The "GUI-process" offers a cooperative interface of IGAV including following objects.

● **Problem-Canvas**

By using IGAV, we cope with constrained optimization problems formulated as:

$$\left(\begin{array}{ll} minimize & g(\mathbf{x}) \\ subject\ to & h_i(\mathbf{x}) \le 0 \quad (1 \le i \le \bar{m}), \\ & h_i(\mathbf{x}) = 0 \quad (\bar{m} < i \le m) \end{array} \right. \qquad (1)$$

where $g : \mathcal{R}^n \to \mathcal{R}$, and $h_i : \mathcal{R}^n \to \mathcal{R}$.
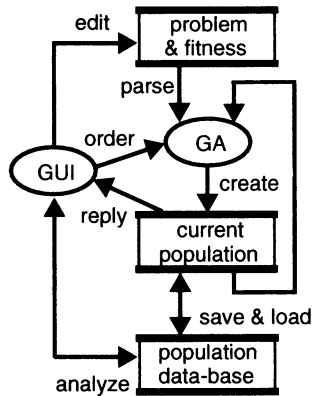
Figure 1: Functional model of IGAV

In order to handle constraints in GA, we use penalty technique and expand constrained optimization problems into unconstrained optimization problems. By using "Problem-Canvas" as shown in Fig.2, we define an arbitrary penalty function, or fitness function $f(\mathbf{x})$, from the functions $g(\mathbf{x})$ and $h_i(\mathbf{x})$ $(1 \le i \le m)$ in (1). Furthermore, we also appoint some coefficients included in the fitness function as variable ones. The values of these coefficients can be changed interactively as well as the GA parameters.
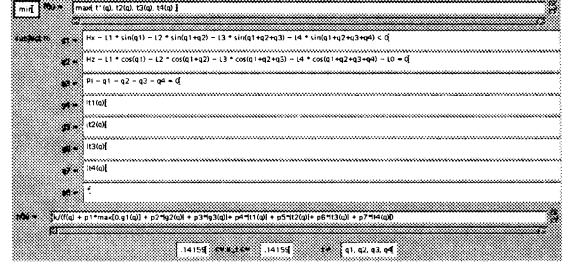
Figure 2: Problem-Canvas

● **GA-Window & Control-Panel**

We observe the progress of population through "GA-Window" as shown in Fig.3. GA-Window displays the fitness $f(\mathbf{x})$ of all members of each population versus the generation number. Therefore, we can easily evaluate the diversity of population. According to the response on GA-Window, we select genetic operations and change the values of the GA parameters interactively. Most of the functions of IGAV are controlled by using "Control-Panel" at the bottom of Fig.3, on which only available menu-buttons appear automatically according to the situations of IGAV. Such an intelligent function of Control-Panel supports effective man-machine cooperation.
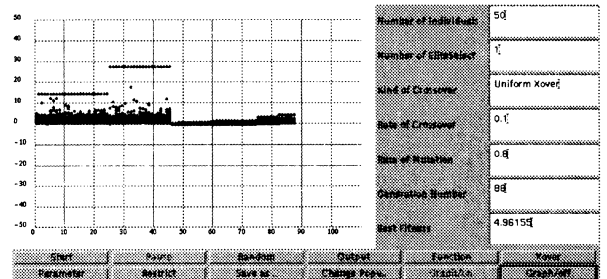
Figure 3: GA-Window & Control-Panel

● Solution-Window

When we deal with constrained optimization problems, it is difficult to evaluate the goodness of solutions from their fitness. Because the best solution is not necessarily a feasible solution which satisfies all of the constraints. Therefore, "Solution-Window" as shown in Fig.4 displays the objective function $g(\mathbf{x})$ and all the constraint functions $h_i(\mathbf{x})$ of the best solution in each population versus the generation number. According to the response on Solution-Window, we can change the values of penalty coefficients included in the fitness function interactively. Furthermore, Solution-Window is also helpful to define an appropriate fitness function $f(\mathbf{x})$ by using Problem-Canvas
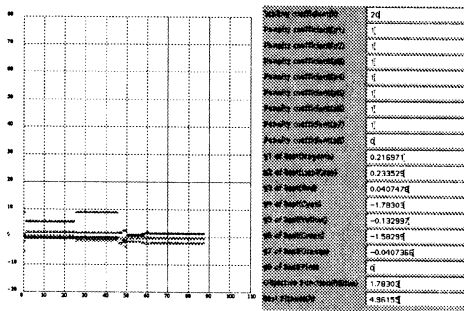
Figure 4: Solution-Window

● Population-Dialog

As you can see in Fig.1, each population, i.e., a set of solutions created by the GA process, is regarded as a file of data. Therefore, the current population can be saved to and/or loaded from the data-base storing various populations at any moment. The populations stored in the data-base can be also analyzed statistically. We manage such a bunch of populations by using "Population-Dialog" as shown in Fig.5.
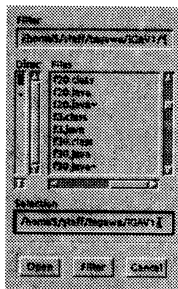
Figure 5: Population-Dialog

3 Design of Optimal Postures

3.1 Mobile Manipulator

In this paper, we consider a five-DOF planar mobile manipulator as shown in Fig.6. The mobile manipulator is composed of a manipulator with four revolute joints and a mobile platform moving along the horizontal axis. The kinematics equations of the mobile manipulator are given by:

$$\begin{cases} x_e & = & x_p + k_1(\mathbf{q}) \\ y_e & = & k_2(\mathbf{q}) \\ \theta_e & = & k_3(\mathbf{q}) \end{cases} \quad (2)$$

where, x_e and y_e represent the position of the end-effector in the absolute reference frame; θ_e represents the orientation of the end-effector; x_p is the position of the mobile platform; $\mathbf{q} = (q_1, \ q_2, \ q_3, \ q_4)^T$ is the vector of joint variables of the manipulator.

Assuming that a point load $\mathbf{f}_e \in \mathcal{R}^3$ is applied at the end-effector, the contributed manipulator actuator torque vector $\tau = (\tau_1, \ \tau_2, \ \tau_3, \ \tau_4)^T$ on the static system can be calculated as:

$$\tau = J^T(\mathbf{q}) * \mathbf{f}_e \quad (3)$$

where $J(\mathbf{q})$ denotes the manipulator Jacobian matrix.

We suppose that the mobile platform is fixed on the ground at a point x_p by enough frictional force.

3.2 Optimization Problem

Designing an optimal posture of mobile manipulator can be formulated as a constrained optimization problem as shown in (1). In order to minimize the torque values of respective actuators, we adopt min-max approach. From the statics equation in (3), we
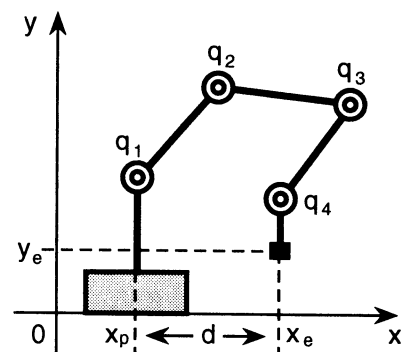
Figure 6: Mobile Manipulator

define an objective function to be minimized as:

$$g(\mathbf{q}) = \max_{1 \leq j \leq 4} \{ \, |\tau_j| \, \} \tag{4}$$

Furthermore, we have to consider some constraints for keeping the position and the orientation of the end-effector. For manipulating the end-effector in safely, we need the following inequality-constraint:

$$x_e - x_p \geq d \tag{5}$$

where d is a lower-bound of the distance between the base of the manipulator and the end-effector.

Because the mobile platform can move along the horizontal axis, we can eliminate the unknown variable x_p that represents the position of the mobile platform. Consequently, we can derive three necessary constraints from the kinematics equations in (2) and the inequality-constraint in (5) as follows:

$$\begin{cases} h_1(\mathbf{q}) & = & d - k_1(\mathbf{q}) & \leq & 0 \\ h_2(\mathbf{q}) & = & y_e - k_2(\mathbf{q}) & = & 0 \\ h_3(\mathbf{q}) & = & \theta_e - k_3(\mathbf{q}) & = & 0 \end{cases} \tag{6}$$

where, $-\pi \leq q_i \leq \pi$ $(1 \leq i \leq 4)$.

3.3 Experimental Results

We have applied IGAV to the constrained optimization problem given by (4) and (6). By using penalty technique, we have defined a fitness function $f(\mathbf{q})$ to be maximized as:

$$f(\mathbf{q}) = \cfrac{a_0}{a_1 * \max\{0, \, h_1(\mathbf{q})\} + \sum_{i=2}^{7} a_i * |h_i(\mathbf{q})|} \tag{7}$$

where $h_{j+3}(\mathbf{q}) = \tau_j$ $(j = 1 \sim 4)$, and the values of a_i $(i = 0 \sim 7)$ can be changed interactively.

Experimental results are shown in Fig.3 and Fig.4. The quality of obtained solutions and the computational time spent for them depend on the abilities of human designers operating IGAV. However, we could confirm that IGAV is useful to obtain desirable postures of mobile manipulators in practical time.

4 Conclusions

We have present a man-machine cooperative design tool, Interactive GA Viewer (IGAV), to use GA interactively and realize the emergent design. The developed IGAV has the merit of solving constrained optimization problems more effectively. Then, we have applied IGAV to the design of optimal postures for mobile manipulators successfully.

In our future works, we would like to implement IGAV over the World Wide Web (WWW), and establish a collaborative design system on the distributed environments.

References

[1] F. G. Pin, Jean-C. Culioli and D. B. Reister, "Using minimax approaches to plan optimal task commutation configurations for combined mobile platform-manipulator systems," *IEEE Trans. on Robotics and Automation*, **10**(1), pp.44–54, 1994.

[2] K. van den Doel and D. K. Pai, "Performance measures for robot manipulators: a unified approach," *The International Journal of Robotics Research*, **15**(1), pp.92–111, 1996.

[3] S. Kitamura, Y. Kakuda and H. Tamaki, "An approach to the emergent design theory and applications," *Proc. of AROB III'98*, pp.77–80, 1998.

[4] Y. Yoshida, N. Adachi and K. Tajima, "GAVIEW - a visualization tool for supporting GA simulations and analyses," *Proc. of IEEE Int. Conf. on Evolutionary Computation*, pp.437–442, 1996.

[5] W. B. Shine and C. F. Eick, "Visualizing the evolution of genetic algorithm search processes," *Proc. of IEEE Int. Conf. on Evolutionary Computation*, pp.367–372, 1997.

[6] T. F. Bersano-Begey, J. M. Daida, J. F. Vesecky and F. L. Ludwig, "A Java collaborative Interface for genetic programming applications: image analysis for scientific inquiry," *Proc. of IEEE Int. Conf. on Evolutionary Computation*, pp.477–482, 1997.

[7] M. Gen and R. Cheng, "A survey of penalty techniques in genetic algorithms," *Proc. of IEEE Int. Conf. on Evolutionary Computation*, pp.804–809, 1996.

[8] J. Rumbaugh, et al., *Object-Oriented Modeling and Design*, Prentice Hall, 1991.

[9] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Third, Revised and Extended Edition, Springer, 1996.

Effects of Population Topology in

Distributed Parallel Genetic Algorithms

Dongcheng Hu   Rui Jiang   Yupin Luo

Dept. of Automation, Tsinghua University, Beijing 100084, P.R.China

EMail: rjiang@mail.au.tsinghua.edu.cn

Abstract

A Distributed Parallel Genetic Algorithm discussed in this paper consists of many groups of population (colonies) that evolve concurrently. Each colony maintains a group of chromosomes and performs certain kind of communication with its neighboring colonies in each generation. Two main aspects in such an algorithm — the communication topology of colonies and the communication intensity among colonies are discussed in detail. Some results are given with simple analysis over them.

Keywords: Distributed Parallel Genetic Algorithm, Communication Topology, Communication Intensity

1 Introduction

The genetic algorithm (GA) constitutes a large class of evolutionary computing methods with many applications in both technological and scientific domains. Its implicit parallelism has been proved to be responsible for its success in many challenging optimization problems which are difficult to solve using conventional techniques[1][2]. Thanks to the ability of exploring different regions of the solution space in parallel and maintaining multiple solutions concurrently, genetic algorithms can execute search tasks quickly with small probability to be trapped into local optima[3][4]. Many researchers such as Tanses[3] and Nang[4] have given their attentions to these properties and made great efforts in extending the implicit parallelism of genetic algorithms. Various coarse-grained parallel genetic algorithms which control a number of colonies to evolve concurrently have been developed under their efforts[3][4]. It is of great interest to consider whether special architecture of these colonies could result in special effects in certain optimization problem[5]. K.Y.Szeto may be the first person that takes steps over this thought. By placing colonies on a hyper-cubical lattice and maintaining them to obtain certain architecture, he gets the result that different architecture will result in different efficiency[5].

The advent of network era has changed our conception on computation. Distributed parallel computing will be the main computing trend in the next century. Therefore, it is not only possible but also indispensable to develop theories of genetic algorithms in order to acclimatize it to distributed parallel computing. We point out that, when a genetic algorithm is distributed into several sites on the network, many new features will generate and several aspects should be considered. Among them, the topology of population (i.e., colonies) and the communication intensity among colonies are the most important ones. Our intention in this paper is to discuss these two aspects and give out some simple results.

The rest of the paper is organized as follows. In section 2, we discuss coordination of populations within distributed parallel genetic algorithms, including communication topology and communication intensity among populations in an algorithm. In section 3, we illustrate distributed parallel genetic algorithms with different topologies by limit communication intensity in the algorithm to a constant. In section 4, the effects of different topologies on our distributed parallel genetic algorithms are analyzed. Finally, in section 5, we conclude the paper and bring forward several aspects, which are worth noting in our future work.

2 Coordination within Distributed Parallel Genetic Algorithms

Slightly different from parallel genetic algorithms in conventional concept, which are built on parallel computers, our distributed parallel genetic algorithm is built upon local network, which is consisted by several workstations. Briefly speaking, our distributed genetic algorithm is a kind of coarse-grained distributed computing methods. It consists of several independent computing units, which are distributed in the concept of physical but can contact with each other since they are connected by network. Independent population (i.e., colony) evolves in parallel in each computing unit. In every computing cycle, some individuals of each colony will be send to others though network according to certain rule in order to keep the all the colonies evolving at the same time.

The transfer of individuals among units is absolutely indispensable to a distributed parallel genetic algorithm, because the diversity of population which is kept by this way is very important to global performance of a genetic algorithm. The essence of this kind of transfer is the coordination among colonies, which is represented by the topology of the communication and the intensity of the communication among colonies. Therefore, the communication topology and the communication intensity are two important aspects need to be studied in our distributed parallel genetic algorithm.

We can incorporate communication topology factor C^t and communication intensity factor C^i in order to describe the algorithm in a quantitative way and define them as:

Definition 1: C^t is the communication topology factor, which represents the number of neighboring units to each computing unit.

Definition 2: C^i is the communication intensity factor, which represents the number of individuals that will be transferred to other colonies by one colony in each generation.

In most cases, we keep both C^t and C^i as constants in each generation, thus the communication topology and intensity are kept stable in the whole evolving process. But we can also change them between generations in order to get special effects.

3 Distributed Parallel Genetic Algorithms with Different Topologies

In this section, we will limit communication intensity C^i to a constant to discuss different effects that result from different topologies.

3.1 Logical architecture and C^t

Suppose our algorithm is distributed into M workstations. In each workstation, there exists an evolution colony consists of N individuals (i.e., chromosomes). Thus the whole algorithm consists of totally MN individuals. From the physical view, these M colonies are distributed in M workstations; from the logical view, these M colonies are organized to build certain architecture (e.g., one-dimensional ring, two-dimensional lattice and three-dimensional cubic lattice). Communication can be performed between every two neighboring colonies. The core conception here is the definition of neighboring colonies. Fig.1 gives us some examples to show neighboring colonies of a colony in one, two and three dimensional logical spaces.
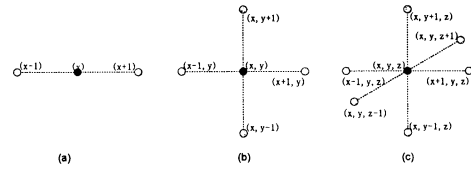
Fig.1 : Neighboring colonies of a colony

In the figure, colony (x) in one-dimensional logical space is neighboring to colony $(x+1)$ and $(x-1)$. Colony (x, y) in two-dimensional logical space is neighboring to colony $(x+1, y)$, $(x-1, y)$, $(x, y+1)$ and $(x, y-1)$. In general, in D dimensional logical space, a colony is represented by $(x_1, x_2, ..., x_i, ..., x_{d-1}, x_d)$ with $x_i \in \{1, 2, ..., M_i\}$. In order to get a symmetric architecture, in which all M_i have the same value M_D, the equation of $M = M_D^D$ must be satisfied. In such a symmetric architecture, neighboring colonies of colony $(x_1, x_2, ..., x_i, ..., x_{d-1}, x_d)$ are defined as:

$$(x_1, x_2, ..., x_i \pm 1, ... x_{d-1}, x_d), \quad (i = 1, 2, ..., d)$$

When $x_i = M_D + 1$, we give it the value of $x_i = 1$. Similarly, we use $x_i = M_D$ instead of $x_i = 0$. Thus, in D dimensional symmetric logical space, a colony has $2D$ neighboring colonies.

3.2 Communication intensity and C^j

In every generation, each colony selects L individuals and sends them to its neighboring colonies for exchanging. Since the colony gets also L individuals from its neighboring ones, the individual number will keep constant to every colony in the whole evolution process. The core factor here is the communication intensity factor C^i, which cannot be set too large or too small. With a too large C^i, there are so many individuals exchanged among colonies in each generation that colonies may not evolution in a stable way. With a too small C^i, communication among colonies is too weak to have effect on the evolution process. Since our purpose is to limit communication intensity as a constant, we choose $C_d^i = 0.2$ ($d = 1, 2, ...$).

Let's consider individuals exchanged between two colonies (referred as n), we get:

$$L = C^t \bullet n \quad \text{and} \quad L = C^i \bullet N$$

Thus, we have:

$$C^t \bullet n = C^i \bullet N \text{ and } n = N \bullet C^i / C^t$$

Corresponding n that is get from these parameters are shown in Tab.1.

Tab. 1 : C^t, C^j, L and n in different dimensions

Dimension	C^t	C^j	L	n
1	2	0.2	$N/5$	$N/10$
2	4	0.2	$N/5$	$N/20$
3	6	0.2	$N/5$	$N/30$

4 Results

In order to test our distributed parallel genetic algorithm on local network, we have designed an experimental system to demonstrate it.

4.1 Design of the experimental system

All the algorithms in our experimental system are symmetric ones. Therefore the equation of $M = M_D^D$ must be satisfied. In order to make it easy for comparison, we keep M be the same value for all dimensions. Thus, if we want to test our algorithm in dimension one, two and three, following equation should be satisfied:

$$M_1^1 = M_2^2 = M_3^3$$

We get the minimum integers which satisfy

above equation are M_1=64, M_2=8 and M_3=4. The logical architectures in 1, 2 and 3 dimensions are shown briefly in Fig.2, in which a one-dimensional ring, a two-dimensional lattice and a three-dimensional cubic lattice are formed, respectively.
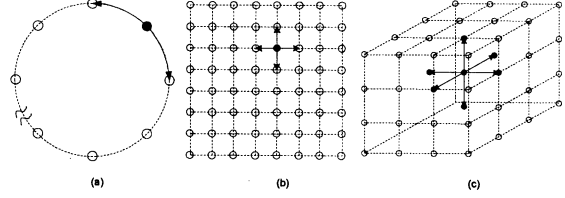
Fig.2 : Different topologies in 1,2 and 3 dimensions

As for the number of chromosomes in each colony, i.e., N, wee can see from the tab.1 that 30 is the minimum value for dimension 1, 2 and 3, but in this case, only 1 chromosome is exchanged between every two colonies. In order to diminish random effect, we choose N=120. With this value, n_1=12, n_2=6 and n_3=4.

As for problem used to test our experimental system, we choose the famous Traveling Salesman Problem (TSP) as test problem.

As for genetic operators used in our experimental system, we use both micro-operators within a colony and macro-operators between every two colonies. Micro-operators include crossover and mutation. For crossover, we choose $N \bullet p_c/2$ pairs of chromosomes out of N chromosomes as parent ones and perform greed algorithm over these two chromosomes to form two better ones. After the operator of crossover, we put child chromosomes into the colony instead of their parent ones. For mutation, we choose $N \bullet p_c$ chromosomes out of N chromosomes and perform two points mutation operator over each of them to form better ones. After mutation, we put newer ones instead of their parent ones into the colony. By now, we use exchange as the only macro-operator. In each generation, after performing micro-operators over each colony, we choose $C^i \bullet N$ chromosomes out of N ones and divide them into C^t groups. Then, each colony transfers these groups to its neighboring colonies, one group for one neighboring colony. Since each colony will receive also $C^t \bullet N$ (for $C^t \bullet n = C^i \bullet N$) chromosomes from its neighboring ones, the total chromosomes in each colony will be kept as a constant in the whole evolving process.

4.2 Evaluation of different topologies

In this sub-section, we will discuss the question of evaluating effects of different topologies on our distributed genetic algorithms. Two parameters are used in our demonstration system to perform this evaluation: one is the average first passage time (T_A); the other is average success ratio (R_S).

Let's first have a look at the average first passage time on a Las Vegas algorithm, which, by definition, always produce a correct solution to a problem but with a distribution of solution times. Suppose that we have M colonies labeled by $i = 1$, $2, ... M$, and their times to find the solution at first time are $\{T_i\}$. This T_i is defined as the first passage time of colony i. By setting time to get the solution as the x-coordination and number of colonies as the y-coordination, we can get a histogram $H(T)$, which satisfies following equation:

$$M = \int_0^\infty H(T)dT$$

In discrete case, the equation is modified to:

$$M = \sum_0^\infty H(T)$$

By defining $P(T)=H(T)/M$, we get:

$$T_A = \sum_0^\infty P(T)T$$

Here, T_A is the average first passage time of the given Las Vegas algorithm.

For a Monte Carlo algorithm, which may fail in finding solutions to the given problem. We can define a maximum acceptable error domain. When the algorithm finds an answer which is in the domain, we take the answer as an acceptable one and make it as the solution to the problem. Thus, we can treat a Monte Carlo algorithm as a Las Vegas algorithm and can use the definition of average first passage time for algorithm evaluation.

As for the parameter of average success ratio for an algorithm, we first define a maximum acceptable error domain and a maximum passage time, which means the maximum acceptable time for an algorithm to find an acceptable solution. Then, we test the algorithm several times (such as $W=5000$), counting trails which get acceptable solution (mark it as W_S). The average success ratio R_S is defined as:

$$Rs = Ws/W$$

In order to compare the efficiency of algorithms with different topologies, we set a common maximum acceptable error domain and get different average first passage time of each algorithm. We also set a common maximum passage time to test different average success ratio corresponding to each algorithm. Experiment results show that under a constant communication ratio, in some extent, a more complex communication topology (a larger C^t) brings benefits for the performance of a given algorithm. The two-dimensional algorithm is better than the one-dimensional one, while the three-dimensional algorithm is better than the two-dimensional one. Results also show that the benefits gained for an algorithm from one-dimension to two-dimension is much more than which from two-dimension to three-dimension.

5 Conclusions

In this paper, we brought forward that communication topology and communication intensity are the most important aspects in a distributed parallel genetic algorithm. By limit communication intensity to a constant, we designed a demonstration system to show that algorithms with different topologies have different efficiencies. In some extent, a complex topology is preferred than a simple one.

In future studies, we will extent our view port to asymmetric architectures, which may refer to fractal dimensional logic spaces.

Acknowledgements

References

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

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

[3] R.Tanese, *Distributed Genetic Algorithms*, Proceedings of Third International Conference on Genetic Algorithm, Lawrence Erlbaum Associates, Hillsdale, N.J., 1989, 434-439.

[4] J.H. Nang, *A simple parallelizing scheme of genetic algorithm on distributed-memory multiprocessors*. International Journal of High Speed Computing, 6, 3(1994), 451-474.

[5] K.Y.Szeto, K.H.Cheung, and S.P.Li, *Effects of Dimensionality on Parallel Genetic Algorithms*, Proceedings of World Multiconference on Systemics, Cybernetics and Informatics.

EcoGA: Maintaining Diversity through Adaption in Genetic Algorithms

Lucia Sheehan, J.J. Collins, and Malachy Eaton
Dept.of Computer Science and Information Systems
University of Limerick,
Limerick. Ireland.
Email: [lucia.sheehan, j.j.collins, malachy.eaton]@ul.ie

Abstract

Traditionally, genetic algorithms (GAs) have required intensive fine tuning of the parameter set to facilitate effective search of the fitness landscape. One solution described in this paper, is to extend the neo-Darwinian paradigm upon which GAs are based, to incorporate mechanisms and operators to facilitate adaptiveness. In doing so, genotypic diversity is maintained, as in natural systems. This allows the application of GAs to a wider range of problems and a less specialised user base. This new GA paradigm has been named the EcoGA.

1 Introduction

Adaptation of parameters and operators is one of the most important and promising areas of research in evolutionary computation, Hinterding, Michalewicz, and Eiben [1]. Reported attempts have adapted various parameters using techniques such as co-evolution or performance-based methods, Smith and Fogarty [2].

This research evaluated a number of these methods and incorporated some which were found to operate well within the EcoGA framework. Further concepts based on biological inspiration were also incorporated in the EcoGA, concepts which have not been previously modelled in a GA.

Incorporating further biological concepts has many advantages such as automatic parameter tuning, varying parameter values during run and allowing the GA to adapt to a changing environment. These advantages will allow the GA to be applied to a wider class of problems including dynamic problems. The adaptive concepts incorporated into the EcoGA are an adaptive mutation rate, life expectancy for each individual, variable population size and an adaptive competition mechanism.

One ongoing problem with GA's is premature convergence, especially when applied to dynamic environments. One method to reduce this problem is to increase the diversity of the population. As long as the search space is well represented in the population, the GA will be able to adapt changes to the fitness function by concentrating it's efforts on the region of space that is now favoured by the changed fitness function - Grefenstette [3]. This also applies to stationary fitness functions particularly in deceptive or multi-modal problems where the algorithm may get stuck on a local optimum and never find the global optimum. Two main factors affect diversity in natural biological systems, natural selection and the reproduction operators such as crossover and mutation. Equally, these are the factors which affect diversity in GAs and are those which have been altered and incorporated into the EcoGA.

2 EcoGA

The EcoGA focuses on incorporating both adaptivity and diversity to achieve the goals outlined above. The following concepts assist in creating a framework which will enhance both these features.

Resources are central to the EcoGA as individuals have to acquire them to reproduce. They are modelled on computer memory which is limited, and allocated on the basis of competition. Competition is well founded in biological organisms who compete daily for resources in order to survive and reproduce. To reproduce, individuals in the EcoGA first have to acquire resources to sustain the initial life of their offspring. The adaptive nature of competition is such that it only occurs when resources are constrained, allowing individuals of low fitness to reproduce. This in turn increases the diversity of the system. Once an organism is created it is added to the population to co-habit with its parents/children etc.

Similar to biological organisms, individuals in the EcoGA will have an expected lifespan. This is cal-
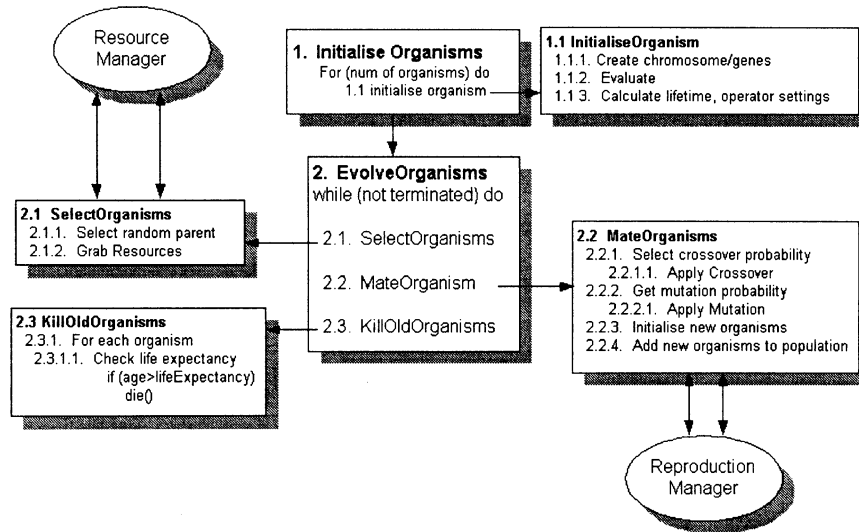
Figure 1: Logical view of EcoGA Algorithm

culated on fitness and is based on work by Arabas, Michalewicz, Mulawka [4]. As the selective pressure from resource competition is relatively low, further selective pressure is incorporated through the adaptive life expectancy mechanism. Death occurs when the organism has exceed its life expectancy, but can also occur due to the effect of the MutateOrDie operator. This is based on biological research which shows that in times of ecological crises, the death and mutation rate of organisms increases, Sniegowski, Gerrish and Lenski [5]. This is incorporated into the EcoGA by firstly specifying a base mutation rate which is applied during reproduction as in the standard GA model, then either mutation or death may be applied when an individual is unable to acquire resources. This mechanism results in a variable mutation rate which adapts in response to changes in the environment of the EcoGA. Together with competition, this mechanism allows the population size to vary during the run of the GA, and also allows each problem domain to determine its own population size, depending on the size of the chromosome. Figure 1 provides a diagrammatic overview of the algorithm.

2.1 Adaptation

Several of the above concepts will result in an algorithm with has many adaptive features. Namely, the population size will have the ability to adapt to changes in its environment. For example if resources become constrained then fewer organisms will have the resources to reproduce and the population will decrease. Those organisms of higher fitness will have a higher probability to reproduce thus encouraging natural selection. The Mutation Rate will adapt also using the concept of the MutateOrDie operator. Again when resources are constrained then the mutation rate is dynamically increased as those organisms unable to reproduce will undergo this operator. This means that the search space is constantly being explored through the use of increased mutation rates, also, as children co-habit the EcoGA with their parents, the danger of losing useful offspring is minimised.

As outlined above, each individual uses an adaptive mechanism to determine its life expectancy, based on its own fitness. This mechanism further promotes natural selection in the EcoGA by allowing those individuals with higher fitness to live longer thus having a greater probability of reproducing.

2.2 Diversity

One of the aims of this research was to create an algorithm which could be used to solve dynamic problems. The maintenance of diversity is considered to be essential in solving these classes of problems. Diversity in the EcoGA is achieved by altering the factors which affect the selection bias. If the selection bias is very strong then the diversity will be low and the algorithm may converge prematurely. On the other hand if the selection bias is very low the diversity will be high but a solution may never be found. By isolating those factors which affect selection bias, that is the selection mechanism and the reproduction operators, then

we may be able to alter the selection bias. In the Eco-GA the organisms are selected randomly to reproduce. However, further selection bias is incorporated when resources are constrained. Therefore the selection bias increases when resources are constrained and is relaxed when they are freely available. The MutateOrDie operator encourages diversity by increasing the mutation rate, but only in the event of constrained resources.

Using the concepts of MutateOrDie, life expectancy and competition and random selection the selection bias is adjusted and in fact will itself adapt over the course of the run of the EcoGA.

3 Tests and Results

To examine its robustness, the EcoGA was tested on a total of twelve different functions, both static and dynamic. The static test functions included De Jong's suite of test functions, Rastrigin's, Schiffel's, Griewangk's and Schaffer's Functions. The dynamic functions run were Goldberg's Knapsack problem and two other dynamic functions which were developed by Ošmera, Kvasnička and Pospíchal, [6]. Each test was averaged over 25 runs. The results are compared with two versions of the standard GA, namely a generational(G-GA) and a steady state (SS-GA). The results for the static and dynamic environments are shown in figures 2 and 3 respectively. The static tests show that the EcoGA slightly outperforms the G-GA in terms of fitness and diversity. However, a clearer difference is seen in the comparison between the EcoGA and the SS-GA. The SS-GA achieves slightly higher fitness but at a huge cost to diversity, while the Eco-GA achieves relatively high fitness while maintaining a high level of diversity.

The results for the dynamic functions show the diversity and tracking error, which is the ability of the GAs to track changes in the environment. This figure shows that the EcoGA outperformed both the G-GA and the SS-GA, both in terms of fitness attained and maintaining a low error rate. The G-GA maintained a high diversity but failed in accurately tracing changes in the environment. The SS-GA performed somewhat better than the G-GA, but well below the level of performance found in the EcoGA. This illustrates the suitability of this EcoGA to dynamic problems.

3.1 Adaptation and Diversity

It has been illustrated that the development of an adaptive GA framework using the concepts outlined
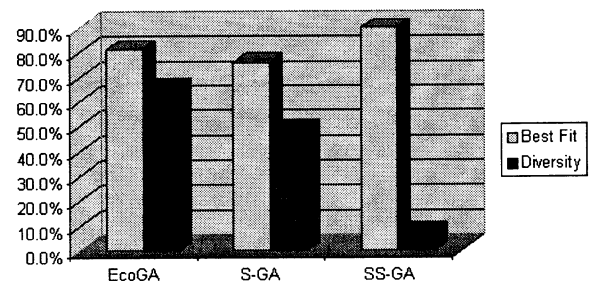
Figure 2: Average error and diversity for static environments
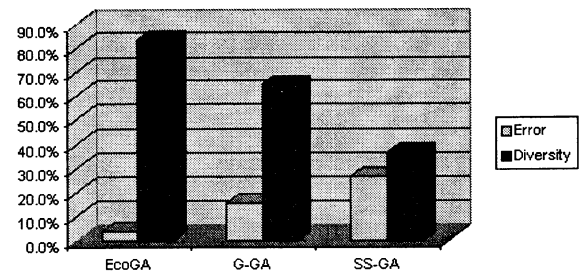
Figure 3: Average error and diversity for dynamic environments

above, the EcoGA is better equipped to solving additional classes of problems. Through this adaptive framework, several parameters were shown to adapt with each problem. For example, the mutation rate has the ability to vary over the course of the run of the EcoGA and can also vary depending on the function to be solved. The figure 4 illustrates how the mutation rate varied when the EcoGA was applied to the static problems.

A similar pattern was discovered when solving the dynamic problems where one of Ošmera et al's problems resulted in a higher overall a higher mutation rate. This problem was considered to be the most difficult in the test suite and performed best when high mutation was applied. This illustrates that the Eco-GA allowed the mutation rate to respond to the needs of the fitness function. It is just such adaptive mechanisms which give this system the power to solve many different classes of problems.

Diversity was largely affected by the inclusion of the competition mechanism and the mutation operator. The competition mechanism ensured that there was sufficient selection pressure to find a good solution. Several tests were run where the competition mechanism was not used. These tests resulted in low

fitness values in the static tests. This would indicate that the lifetime mechanism alone is not enough to ensure a bias towards the best fit and the additional bias given by competition is needed. However, in the case of one of the dynamic functions (Ošmera et al's first function), the exclusion of competition increased the overall fitness of the run. Under further investigation it was discovered that the higher the mutation rate with this function, the better the performance. Also, further tests showed that excluding competition increased the frequency of mutation. Further tests were carried out which increased the base mutation rate and included the competition mechanism. This resulted in improved performance for that fitness function without adversely affecting the performances of the other functions.
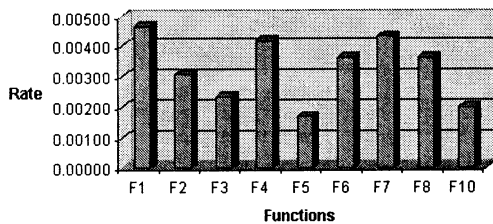
Figure 4: Average mutation rates for each function

4 Discussion & Further Work

The results to date demonstrate the ability of the extended haploid paradigm, namely the EcoGA, to solve dynamic problems with the same effectiveness as diploid genotypes. The EcoGA does this by maintaining diversity through adaptation, by incorporating concepts from natural systems such as constrained resources and lifetime evaluation. The results for the static environments demonstrate that the EcoGA avoids premature convergence and local optima, again through the maintenance of diversity.

To date the EcoGA has been compared to a standard and a steady-state GA, it is our intention to perform further comparisons with other adaptive systems. One problem with performing these comparisons is assessing standards to enable fair comparisons. For example diversity in this study is measured genotypically, while other systems can use different measurements to calculate diversity, for example using a phenotypic diversity measurement. Clearer standards have to be set out and evaluated in order to enable benchmarking

of different types of systems.

This research began with creating a system to solve static problems and went on to apply the same algorithm to solving dynamic problems. While these types of problems are useful in establishing the effects of the EcoGA, ultimately it is desirable to use the system to solve real world problems. As the real world is typified by its complexity and dynamic behaviour, a system which can perform in such situations is desirable. For example, industrial applications such as scheduling and constraint based problems where available resources may vary over time or robot navigation where the environment may change as the robot navigates its way through it. It is our intention to apply this new algorithm to these types of problems. The EcoGA has shown that though the use of its adaptive framework, it can adapt parameters such as mutation rate and competition levels to suit the problem it is solving. It is just such attributes in a problem solving algorithm which will allow it to develop as a problem solving tool to be applied to both old and new problems, all with potentially different needs in terms of parameter settings, diversity and selection bias.

References

[1] Hinterding R., Michalewicz Z., and Eiben A. (1997) Adaptation in Evolutionary Computation: A Survey. In Procs. of the 4th IEEE Int. Conf. on Evolutionary Computation, pp. 65-69.

[2] Smith, J.E., Fogarty T.C., (1997). Operator and parameter adaptation in genetic algorithms. In *Soft Computing 1*, (1997), pp 81-87.

[3] Grefenstette J.J. (1986) Optimization of control parameters for genetic algorithms. In *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16(1):122-128.

[4] Arabas J., Michalewicz Z., Mulawka J. (1994). GAVaPS - a Genetic Algorithm with Varying Population Size. In Procs. of the 1st Int. Conf. on Evolutionary Computation, v.1, pp. 73-78.

[5] Sniegowski P.D., Gerrish P.J. and Lenski R.E. (1997). Evolution of high mutation rates in experimental populations of E. coli. In *Nature*, June 12, 1997; 387(6634):659, 661-2.

[6] Ošmera P., Kvasnička V., Pospíchal J., (1997). In Procs. of the 3rd Int. Mendel Conf., Czech Republic, pages 111–116.

Design of Autonomous Mobile Robot Action Selector based on a Learning Artificial Immune Network Structure

Dong-Je Lee	Min-Jung Lee	Young-Kiu Choi	Sungshin Kim
Dept. of Electrical Eng.	Dept. of Electrical Eng.	Dept. of Electronics Eng.	Dept. of Electronics Eng.
Pusan Nation University	Pusan Nation University	Pusan Nation University	Pusan Nation University
Pusan, 609-735, Korea	Pusan, 609-735, Korea	Pusan, 609-735, Korea	Pusan, 609-735, Korea
dongje@hanmail.net	mnjlee@hanmail.net	ykichoi@pusan.ac.kr	sskim@pusan.ac.kr

Abstract

This paper addresses the construction of the behavior control of an AMR based on artificial immune network. The information that comes from sensors is used for antibody, and the antibody stimulates antigens under environment change. There are many possible combinations of actions between action-patterns and external stimulus. The question is how to handle the situations to decide the final action. We propose a learning artificial immune network to tackle the above problem. The proposed off-line learning method produces antibody automatically. The antibody is generated based on the arbitrary environment and the performance evaluation of the selected behavior. The computer simulation for an AMR action selector shows the usefulness of the proposed learning artificial immune network.

Keywords : Immune network, Action selector, Mobile robot

1. Introduction

The existing AMR's action selector may be classified into two kinds of methods. First, it is to set the whole relations in the form of the modeling to take advantage of the relative formula for achieving a purpose mathematically. Second, it is an artificial intelligence method which imitates, expresses, and implants a human-being's judgement process. The former is a central-concentration system and consists of a simple action-pattern by the set rule. The disadvantage of this system is that it forms the unlimited loop's action-pattern to be in the condition which can't reach to an objective, upon the its entrance into a local loop. Additionally, a numerical formula can be more complicated as an objective becomes diversified and an activity is complicated. The latter is a distribution system, therefore the complex numerical formula is not needed. Also, it can overcome the disadvantage mentioned above. Among the algorithms on the design of an action selector with an artificial intelligence system, there are methods taking advantage of a fuzzy, the neural network, and a genetic algorithm. One of those methods, there is

an artificial immune network which is employed in research. It is to imitate a human being's immune system then apply it to the engineering fields which has been currently in research[1]. The system applying to the AMR's action selector by an artificial immune network algorithm has been achieved[2-4]. This approach is based on the modeled process of a human being that is in excited by an artificial immune network's stimulation and suppression relation. Therefore, it has shown the desirable performance on the activity's judgement. However, the disadvantage is to have the complicated structure and to take long time for implementing an algorithm. Especially, upon the design of an existing action selector, it has a disadvantage to generate the complicated information for antibody by a trial and error.

This learning method is to organize the antibody information more conveniently not through a user, but through a computer's learning algorithm. Also, this paper demonstrates a good performance comparing with the information organized by a user. Finally, the action selector is introduced to show a various of action-pattern's activities through the learning antibody information, through computer simulation and test.

2. The human being's immune system

2.1 Human being's immune system.

The basic structure of a human being's immune network is consisted of a T-lymph and a B-lymph. There are approximately 10^7 of other kinds of B-lymph in a human body. As shown in Figure 1, an antibody is a Y-shaped and is a paratope and an ideotope which has their original information. An artificial immune network by a N. K. Jerne's hypnosis is suggested[5-7].
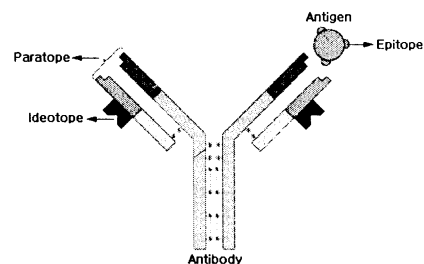
·Fig. 1. Structure of antibody and antigen

2.2 The creation of an antibody.

The human being's immunity is classified into an innate immunity which is possessed by a human being at birth and an acquired immunity which is obtained through life.

3. Action selector by an artificial immune network.

3.1 The design of an action selector.
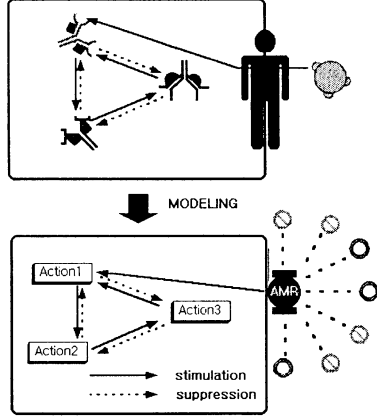
Fig. 2. The modeling of an artificial immune network action selector

Figure 2 shows the action selector's structure of an AMR to imitate the structures of an antibodies which are against the external antigen, a human being's immune system.

3.1.1 Antigen and antibody's information

The recognizing distance is to be the 2 times longer than the AMR's radius as shown in Figure 3. The AMR's active entity is to be limited by 8 such as forward, backward, right $30°$, $60°$, $90°$and left $30°$, $60°$, $90°$.

Each paratope, ideotope, and epitope has the same structure as shown in Table 1.
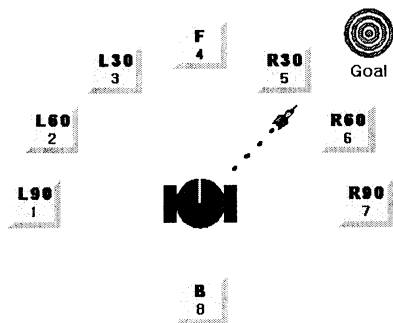
Fig. 3. External environment sensing of AMR
Table 1. The definition of a paratope, ideotope, and epitope structure.

Obstacle sensing (Exist : '1', Don't exist: '0' ,Don't care: '#')								Goal direction ('1~8'or '#')
L90	L60	L30	F	R30	R60	R90	B	Goal

F: front direction   L: left direction
B: back direction   R: right direction

3.1.2 Concentration equation of antibody

The following equation presents the transitive density equation on the i-th antibody[2].

$$\frac{da_i}{dt} = \left\{ \frac{\sum_{j \neq i}^{N} w_{j,i} \cdot a_j}{N} - \frac{\sum_{l \neq i}^{N} w_{i,l} \cdot a_l}{N} + m_i \right\} \cdot a_i \quad (1)$$

$$1 \leq i \leq N$$

$$w_{j,i} = \sum_{k=1}^{L} I_j(k) \oplus \overline{P_i(k)}$$

$$w_{i,l} = \sum_{k=1}^{L} I_i(k) \oplus \overline{P_l(k)}$$

$$m_i = \sum_{k=1}^{L} E(k) \oplus \overline{P_i(k)}$$

where, N is the total number of antibodies and $w_{j,i}$ are in j, i times of a $N \times N$ matrix. m_i is in i times of a $N \times 1$ matrix. L is the size of an antibody's whole bit number and $I_j(k)$ is a bit value in k times of an ideotope of the antibody in j times. $P_i(k)$ is a bit value in k times of the epitope of an antigen. $\oplus$ is an exclusive logical sum operator and has the results '1' and '0'. Upon the operation with '#', the result will be '0'.

The organization of an action selector consists of two kinds of immune networks against a goal and an obstacle, $a_i^{obstacle}$ and a_i^{goal}. γ is an immune network's connective coefficient and the whole immune network's concentration value, A_i is presented as the following equation[2].

$$A_i = (1 - \gamma) \cdot a_i^{obstacle} + \gamma \cdot a_i^{obstacle} \quad (2)$$

$$0 < \gamma < 1$$

The antibody which reaches to the first-established threshold value λ will be selected in terms of the final antibody selection.

3.2 Learning an action selector.

As same as the fact that there are an innate immunity and acquired immunity in terms of a human being's immune system, the shadowed part in Table 2 and 3. presents an innate immunity whose value is that a user can easily think and fixed. The remaining part is an acquired immunity that one of three values such as '1', '0',

and '#', is evaluated and selected by a suitability function on an AMR actions through the random coding information.

Table 2. Structure of an antibody for obstacle avoidance

Antibodies	Paratope for obstacle avoidance								Actions
Antibody 1				0				#	Front
Antibody 2								#	Back
Antibody 3					0			#	Right 30
Antibody 4						0		#	Right 60
Antibody 5							0	#	Right 90
Antibody 6			0					#	Left 30
Antibody 7		0						#	Left 60
Antibody 8	0							#	Left 90

Table 3. Structure of an antibody for goal approach

Antibodies	Paratope for goal approach								Actions
Antibody 1				0				4	Front
Antibody 2								8	Back
Antibody 3					0			5	Right 30
Antibody 4						0		6	Right 60
Antibody 5							0	7	Right 90
Antibody 6			0					3	Left 30
Antibody 7		0						2	Left 60
Antibody 8	0							1	Left 90

The immune network's connective coefficient on the obstacle and goal, γ is to be decided in the paratope learning algorithm. An epitope is fixed matrix. The function for evaluating the suitability after a learning is likely the following.

$$ J = \frac{1}{0.2 + s_1 \times crash + s_2 \times arrive_{distance} + s_3 \times (T_n / T_f)} \quad (6) $$

where s_1, s_2 and s_3 present a plus integral number weight and T_n presents the maximum time which the AMR for a learning can move. crash has '0', upon the crash, and '1', upon the non-crash. $arrive_{distance}$ presents the distance between the AMR location and the goal at final time. T_f is a time that the AMR takes to reach the goal.
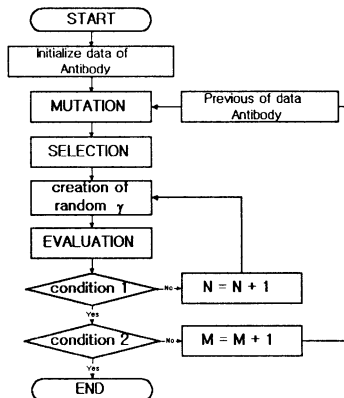
Fig. 5. Flowchart for learning antibody information

The N is the number of generations in order to find out the optimized value of γ. The M is the number of generations in order to find out the optimized information on the antibody's internal information in Table 2 and 3. The condition 1 is for the optimized γ. The condition 2 is for the optimized antibody information.

4. Result of learning by a simulation

4.1 The condition of a simulation

GUI environment is organized in the window 95 for the AMR simulation by an immune network as shown on the Figure 6. The environment for the AMR's move is consisted of the circular obstacles and goals inside 340 $\times$ 190[Cm]. The AMR's moving speed is to be given by 2 [Cm/sampling time]. The AMR's sensing distance is 2 times longer than a robot's radius. No consideration for the time-delay caused by the momentary activity transition, is taken. The internal values for the simulated test are M=10, N=10, T_n =500, s_1=200, s_2=1, s_3=1, γ is mean 0.5 and standard deviation 0.4.
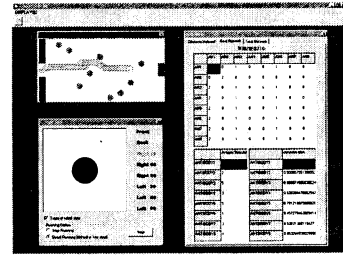
Fig 6. Simulation environment

4.2 The comparison and learning result by a simulation.

Figures 7-10 present the arrangement of an obstacle every 30 times randomly and demonstrate the reaching probability.

4.2.1 An action selector by the set rule.

The Figure 7 shows an action selector by the set rule for the repeated activity.

4.2.2 The learning result on an action selector through the artificial immune network.

The different action-pattern's result according to the obstacle's arrangement is based upon the application of a learning algorithm. Figures 8 to 10 show the various action-pattern's results for the different obstacle's arrangement.

The result of the organized antibody information through a trial and error is shown in Figure 8. It takes an advantage of an artificial immune network and shows 5 repetitions that are displayed in a thick line. It demonstrates the result of getting out of a local loop.
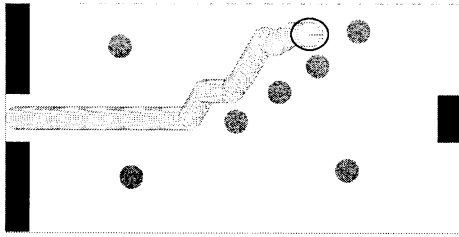
Fig. 7. Result under repeated actions
(Reaching probability ≈ 50%)
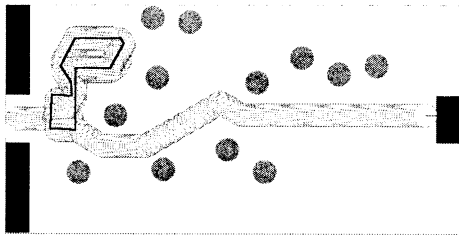
Fig 8. Result of action selector using try and error
method (Reaching probability ≈ 70%)
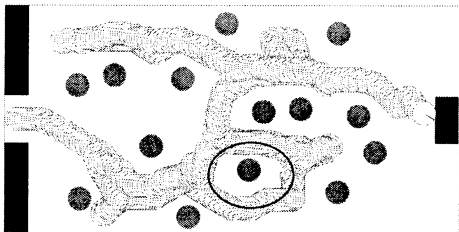
Fig 9. Result under non-repeated actions
(Reaching probability ≈ 70%)
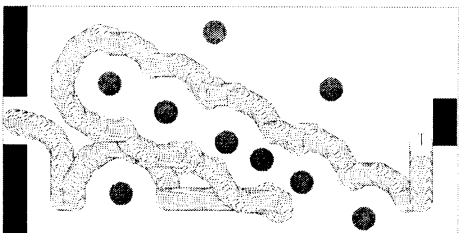
Fig. 10 Result under actions similar to the right-hand-
law (Reaching probability ≈ 80%)
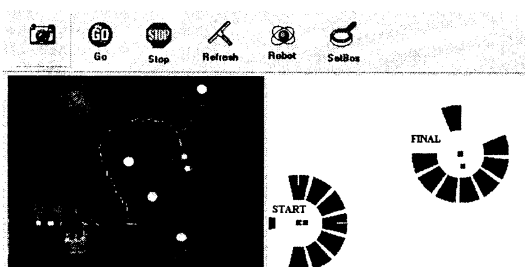
Fig. 11. Result of experiment display

Figure 9 presents the activity of heading for the goal toward the whole areas after having 6 repetitions that are shown in a thick line. In case of the artificial immune network, even though there are, it is not allow to move around the same path to get out of a loop.

The Figure 10 shows the action which is similar to the general right-hand-law of an obstacle-avoiding algorithm. However, it shows the better performance due to the acquisition of advantages such as right-hand-law's algorithm and artificial immune network.

5. Result of experiment

The extensive information is not needed on the actual test. The experiment environment is organized in order to have a convenience on the information's analyzing with a vision sensor. The left part of Figure 11 is a synthesized picture of the first location and the last location of the AMR. The right part is a screen organizing the neighborhood information. The action-patterns used in this test take an advantage of the learning action-pattern with the right-hand-law in Figure 10, to test.

6. Conclusion.

The highly reaching probability is demonstrated in case of using the immune network and applied to the AMR's action selector. The results of Figure7 to 8 demonstrate approximately 70% of the reaching probability on the usage of the trial and error. However, the results of the proposed algorithm present more than 70% of the reaching probability on the usage of the artificial immune network. Additionally, our method shows the advantage of not falling into a local loop again in the immune network. It is possible to design the various action-pattern's AMR action selector with the transition of a learning antibody.

Reference

[1] J. Doyne Farmer and Normal H. Packard, "The immune system, adaptation, and machine learning," *Physica*, 22D(1986), pp. 187-204, North-Holland, 1986.

[2] Akio Ishiguro, Yuji Watanabe and Yoshiki Uchikawa, "An immunological approach to dynamic behavior control for autonomous mobile robots," *IEEE International Conference on Intelligent Robots and Systems*, Aug., 1995.

[3] Akio Ishiguro, Yuji Watanabe and Toshiyuki Kondo, "Proposal of decentralized consensus-making mechanisms based on immune system," *in Proceeding of International Symposium on Artificial Life and Robotics*, pp.122-127, 1996.

[4] Akio Ishiguro Toshiyuki Kondo and Yuji Watanabe, "An immunological approach to dynamic behavior arbitration for autonomous mobile robots," *in Proceedings of International Symposium on Artificial Life and Robotics*, pp. 132-137, 1996.

[5] N. K. Jerne, "The immune system," *Scientific American*, vol. 229, no. 1, pp. 52-60, 1973.

The Clear Explanation of Different Adaptive Behaviors between Darwinian Population and Lamarckian Population in the Changing Environment

Kazuko Yamasaki , Masuteru Sekiguchi
Department of Information Science
Tokyo University of Information Sciences
Tiba ,265-8501,Japan,yamasaki@rsch.tuis.ac.jp

Abstract

Characteristics of population under the dynamic environment for the evolving learning system are studied. The Sasaki[1]showed the differences between the Darwinian and Lamarkian type population for the first time . We examine it in simple "The Continuous Function Model" in order to investigate the mechanism of the phenomenon. Then we obtain the following results and mechanisms are briefly explained.

The Darwinian type population is excellent in the adaptability, when the environment discontinuously changes, and the Lamarkian type population is excellent , when the environment continuously changes. If the same environment is continuing long ,the adaptation of the Darwinian population under the dynamic environment breaks. That is the fitness may dive suddenly because of overadaptation.

And, the experiments are also done in "The Food and Poison Molel" equal to [1], and these are confirmed.

1   introduction

Generally ,two different level of adaptations take place in population of creatures or societies. The one is individual adaptation (learning) within the lifetime of an organism, the other is mass adaptation (evolution) over history of life on Earth. As the interaction of these processes, the Baldwin Effect is well known. From a engineering or sociological point of view , not only Darwinism but also Lamarckism are meaningful. And such systems frequently can be seen in the changing environments , moreover the changes are crucial to some of them, especially social and economical systems . But many studies in the changing environments have not been done ,then large important field remains uncertainly. Recently T.Sasaki and M.Tokoro et all[1] opened the door to such systems. They described that Darwinian population not only showed stable behavior against the oscillation of the environmental condition

and adaptability ,but also maintained a adaptational potential even toward completely new (unknown) environment.But we thought their explanation of these phenomena was not sufficient and further studies were required . We perform the simulation of simple model, "The Continuous Function Model", in the wide variation of periodically changing environment . Then we observe the following results and they are explained clearly by the path in the phase space.

1. Under discontinuous changes of environments, Darwinian population shows stable behavior in wide range of parameter.

2. But when changing period or velocity of evolution becomes large , the average fitness of Darwinian population sometimes falls suddenly. It can be seen also in case of random period.

3. Under continuous changes of environments, Lamarckian population shows rapid adaptation and stable behavior.

We also perform "The Food and Poison Model" which is the same as the Sasaki's model. (Yamasaki[2][3]) In this model , continuous changes can not be simulated because agents learn bit patterns and the paths of phase space can not be followed because the dimension of phase space is large. But we emphasis that observing the same result 1.,2. especially parameter dependence of result 2. being same, means the same mechanism which we describe causes the phenomena of "The Food and Poison Model".

2   The Continuous Function Model

2.1   model

The population is made by 100 agents. Each agent has neural network which is consist of 1 input,2 middle and 3 output layer , and has 7 real number of weights as

Figure 1: environments and output of agents
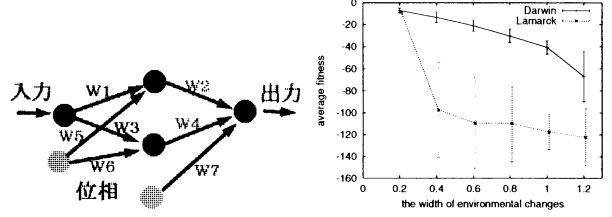
Figure 3: fitness and the width of the change c
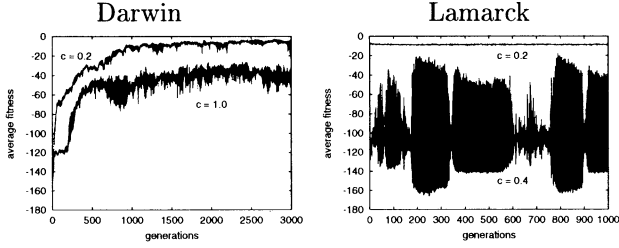
Figure 2: fitness of various changes

Figure 4: Sudden dive of fitness and period and velocity of the evolution

genes.(Fig. 3left) The environment presents the real number of $-1.0 <= x <= 1.0$ in each agent.Each agent estimates function value for input x by the neural network. The environment returns the right function value for each agent. This functional type is changed periodically.

2.2 the result of experiment

- [RESULT 1]For the discontinuous changes of environment Darwinian population stably adapts within the wide parameter range
  The environment is supposed to be one of the functions of the gaussian type $exp(-50(x \pm \frac{c}{2})^2)$ and it takes alternately A environment and B environment every generation.(Fig. 1)(Fig. 2) The relation between mean fitness value of population and the width of the environmental change c is shown in (Fig. 3right). The Darwinian population can stably adapts in the case c is under 1.0. The Lamarckian population can stably adapts in the case c is 0.2 but stability is lost in 0.4.

- [RESULT 2] Sudden dive of fitness can be observed
  Next, the experiment is carried out that the period of the environmental change is changed to stably adapted Darwinian population. As (Fig. 4left) shows,in the case in which the period is lengthened or case in which the period is randomized, the fitness often drops suddenly. Next,we

examine how this sudden dive depends on the velocity of the evolution and period of the environmental change. As the parameter which controls the velocity of the evolution we adopt the width of the mutation , and as the quantity which quantitatively shows the sudden dive we use mean value of decrease width which is the mass average fitness decrease of the every environmental change. (Fig. 4right) shows that with the increase of velocity of the evolution , with increase of the period, the decrease width enlarges.

- [RESULT 3]For the continuous changes of environment Lamarckian population stably adapts
  In order to bring environmental change close to the continuous variation, we carry out the experiment in the model that x_0 of $exp(-50(x-x_0)^2)$ is subsequently making rounds n values from $x_0 = -\frac{n-1}{2}c \sim \frac{n-1}{2}c, c = \frac{1.6}{n-1}$. (Fig. 5) As the number of the environment n increases , Darwinian population becomes bad the adaptation, and Lamarckian population improves the adaptation.

2.3 the mechanism of adaptation

We demonstrate the reason why the phenomenon can be observed by space between 7 weights of the neural network (the adaptation space). The locus from birth to death of 100 agents in two generation are described in the four 2_dimensional graphs,W1-W3,W5-W6,W2-W4 and W5-W7.
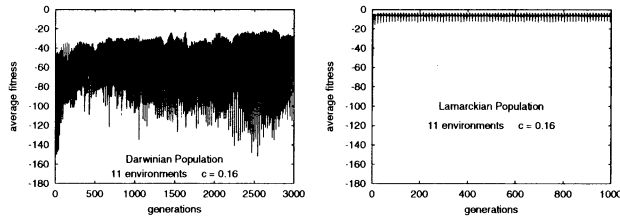
Figure 5: fitness and the continuous changes of environment
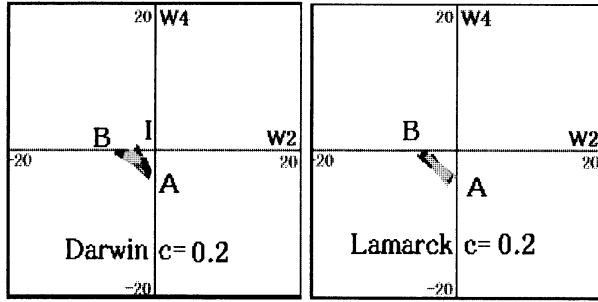
Figure 6: the locus of the weight spaces

- [REASON 1]Why dose Darwinian population stably adapt within the wide parameter range for the discontinuous changes of environment?

c=0.2 both populations adapt
In (Fig. 6), we illustrate 100 agents' locus respectively over 2 generations. (A environment-deep color locus and B environment-light color locus) Darwinian population dose wedged motion I → A I → B , and Lamarckian population dose reciprocating motion A ↔ B. It is mainly remarkably seen in the W2-W4 space. Here I:initial values set, A(B): final values in A(B) environment.

c=0.4 Lamarkian population becomes un-
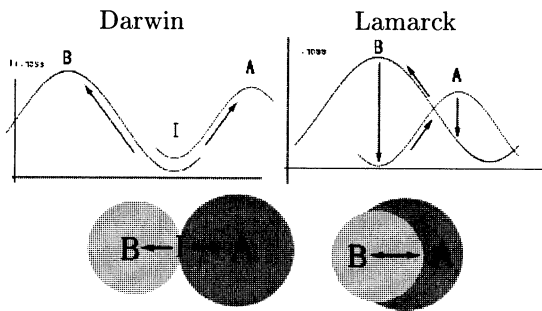
Figure 7: hill climbing and the discontinuous changes of environment
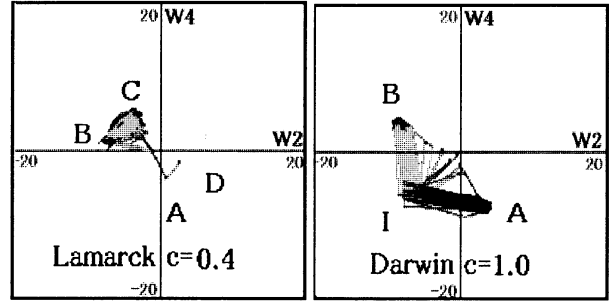
Figure 8: the locus of the weight spaces

stable
(Fig. 8left) shows that Lamarckian population cannot do reciprocating motion A ↔ B, then it does reciprocating motion B ↔ C, which is not adaptable.

c=1.0 only Darwinian populations adapts
(Fig. 8right)shows that the wedged motion of Darwinian population is observed more clearly .

In order to explain these result, we make the following consideration. In (Fig. 7), site A(B) of the weight suitable for the A(B) environment and circleA(circleB) of the attraction area for site A(B) are shown. In the Lamarkian population , like (Fig. 7), it is necessary that circleA and circleB pile up over at least more than the half , in order to be learned by hill climbing A → B , B → A. In the Darwinian population ,if only circleA meets circleB at one point , if the site I moves to this point by evolution, it becomes possible to be learned by hill climbing I → A , I → B. The estimate volumes of adaptable area are as follows.

Volume of Darwinian population $< (2\lambda)^n$
Volume of Lamarckian population $< \lambda^n$

Here,λ :diameter in attraction area and n :the dimension of the adaptation space(number of the weights)
The differences become crucial when n become large, and in usual experiment and general organism level , n is large. In other words ,the learning is possible when "set of multiple environments" are located in the closer position than boundaries of such area, and area volume of Darwinian population is definitely larger than Lamarckian's, and then,they learn generality of "set of multiple environments" .

- [REASON 2]Why is sudden dive of fitness ob-
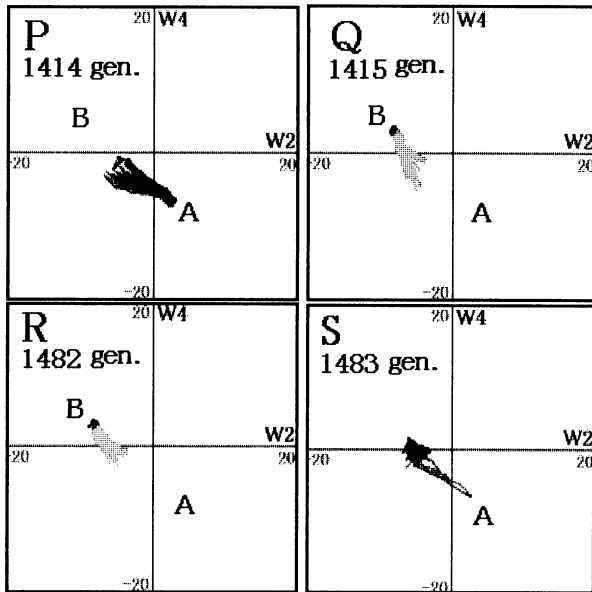
Figure 9: the locus of the weight spaces around the dive

Figure 10: the locus of the weight spaces in the continuous changes of environment

3 Summary

It is interesting that we have often met in our society or history, the overadaptation destroying oneself. We believe this evolving and learning system is regarded as a prototype of the human institution, not relating us in detail of it. In the future work, we will describe the human institution awaking various adaptive mechanisms alternately according to the dynamic changes of the environments .

Furthermore the dynamic changes of the environments described here are not sufficient. Because in the society, the things being learned are not exogenous but endogenous. We have to consider strong interaction among the agents under the dynamic changes of the environments .

We thank other members of Tokyo University of Information Sciences for useful discussion.

served? Why do the increase of period of environmental change and the increase of velocity of evolution promote this dive ?

(Fig. 9)shows W2-W4 space at P,Q,R,S which are described in (Fig. 4left).At P (1414 generation), it normally adapts to the A environment. It changes in to the B environment at Q (1415 gen.) . Since the B environment continues, the values of the weights which are heredity from the parents move over the about 60 generation till R (1482 gen.) according to the Baldwin Effect . It is not already possible to adapt, when it changed in to the A environment at S (1483 gen.) . Because this Baldwin Effect has the positive effect for period of the environmental change and velocity of the evolution ,the increase of period of the environmental change and the increase of velocity of evolution promote this dive.

- [REASON 3]Why dose Lamarckian population stably adapts for the continuous changes of environment?
  (Fig. 10) shows the locus of the both populations in the model that x_0 of $exp(-50(x - x_0)^2)$ is subsequently making rounds 11 values from $x_0 = -5c \sim 5c, c = 0.16$. Lamarckian population can smoothly move over 11 environment continuously, but Darwinian population cannot widely move as the value of initial weights become an anchor.

References

[1] Sasaki,T. & Tokoro,M.: Adaptation toward changing environments Why Darwinian in nature?: *In 4th European Conference on Artificial Life* pp.145-153,1997:

[2] Yamasaki,K. & Sekiguchi,M.: Characteristics of Evolving Learning System and Mechanism of Adaptation under Dynamical Environment *the 8th Workshop on Multi-Agent and Cooperative Computation 1999*

[3] Yamasaki,K. & Sekiguchi,M.: The Emergency of Various Adaptive Mechanisms in the Evolving Learning System under Dynamic Environment *Workshop on Intelligence and Complex System 2000*

PICO_SCAN - using body data to create artificial life forms

by Laurent Mignonneau (1) & Christa Sommerer (2)

(1) & (2) ATR Media Integration and Communications Research Laboratories
2-2 Amity Hikaridai, Seika-cho, Soraku-gun, 61902 Kyoto, Japan
Tel: 81-77495-1426, Fax: 81-77495-1408
laurent@mic.atr.co.jp, christa@mic.atr.co.jp, http://www.mic.atr.co.jp/~christa

Keywords

Artificial Life, User Interaction, Interface, Body Data, Interactive Art

Abstract

Artificial Life as a field of research attempts to synthesize life *in silico* by using computers to create virtual life. By definition life and artificial life should display the following characteristics: self-organization, metabolization, self-reproduction, and adaptive evolution. Until now, most Artificial Life inspired artworks have created artificial characters or creatures that display some kind of behavior but do not really feature all of the above characteristics. The modeling of artificial evolution is a major challenge in Artificial Life research as well as for Artificial Life artworks. In the past we have developed several interactive computer art systems that use artificial life principles in combination with user-machine interaction. [1] The underlying aim of these systems is to study the application of Artificial Life principles to the creation of self-sustaining and evolving interactive artworks. The interaction of the audience with these works has a significant impact on the evolution of the works; by linking the interaction parameters of the users' interactions to the evolutionary software structure of the system, we aim to create artworks that can interpret and visualize the users' interaction with these works and furthermore enable adaptive evolution within the works.

1. Introduction

To capture the users various interaction parameters and to link them to the evolutionary image processes of the works we often produce custom designed interfaces. These have so far included living plants [2], a drawing input device [3], light [4], a gesture recognition system [5] and a text input device [6]. While we have used one specific detection interface for each of these systems, we recently designed a new system called PICO_SCAN, that aims to capture various body data of the user and link them to the creation and metabolism of artificial life creatures. The following sections will describe the whole system in more detail.

2. PICO_SCAN: system overview

The PICO_SCAN system consists of
- a PICO_SCANNER interface device
- a 42 inch plasma video screen
- a video key mixer that can mix video and CG images

Fig. 1 PICO_SCAN system overview

The system overview is shown in Figure 1. PICO_SCAN is designed to be very easy in its use: as the user scans

along her own body she generates various input data that are specific to her own body. The collected data information is then used by the system to generate artificial life creatures that can feed on these data. Ultimately the aim is to create an artificial life environment where the creation, metabolism and evolution of these creatures is linked to the users individual interaction parameters.

2.1. PICO_SCANNER: a sensing system to capture body data

The PICO_SCANNER interface device consists of various sensors combined into one unit. These sensors are:
- 1 lipstick color video camera
- 1 colorimetry sensor
- 1 touch sensor
- 1 3-D position sensor (Polhemus)
- 1 distance sensor
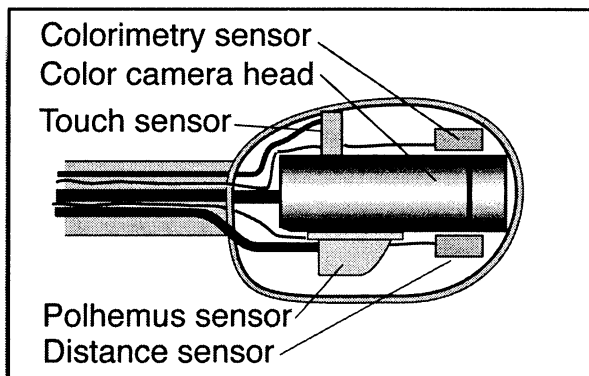
Fig. 2 PICO_SCANNER

Figure 2 shows the PICO_SCANNER which measures about 10 cm in length. When the user picks up the device and scans along her body she generates input data such as distance values, 3 D position values, color and colorimetry values as well as a video image. All data are voltage values that can be converted into digital values to be used for further calculations by the host computer. In the case of PICO_SCAN we use the digitally converted voltage values for the creation of artificial life creatures.

2.2. PICO_SCAN's video mixer

A specifically designed video mixer allows us to key between the video image captured by the lipstick camera and the computer generated (CG) artificial life creatures. Figure 3 shows an example of this process.
When the users holds the PICO_SCANNER at a distance of around 40 cm to her body, the device only captures the users video image. But when moving closer, the device generates images of artificial life creatures that are gradually mixed into the video image. This is done step-less through our in-house video mixer. Figure 3 shows how

the user can switch between the different keying modes. The 3-D position sensor (Polhemus) and the distance sensor provide the necessary position data for calculating the distances between the users body and the PICO_SCANNER.
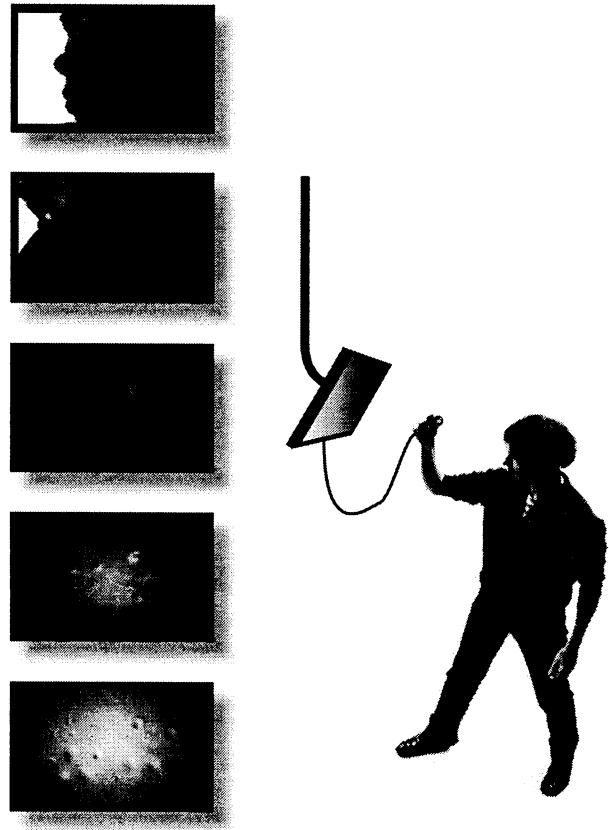
Fig. 3 Switching between video and/or CG images

3. Creating artificial life through interaction

While most artificial life simulations are closed systems [7, 8, 9, 10] we aim to link the real world data of the users' interactions to the virtual world data of the artificial life creatures. To do so we use the color, colorimetry, distance, 3-D position and touch values captured by the PICO_SCANNER. In contrast to our previous systems where the creation process of creatures was directly mapped to the interface input data [11, 12], we start this time with a "soup" of random creatures. When not interacted with these virtual creatures exist but do not move or metabolize; they are inert and merely exist in the memory space of the host computer. Their condition could be compared to "hibernation." When the user picks up the PICO_SCANNER these creatures start to wake up and move. While scanning along her body different body data as well as a video images are being generated that influence the behavior of the artificial creatures.

4. Metabolism, Reproduction and Evolution

In the stage of "hibernation" a creature does not move or metabolize. As soon as activated it will start to move and consume energy. A creature's behavior is basically dependent on two parameters: a) its Energy level (E)
b) its Speed (S) or ability to move

4.1. Energy Level

While the Speed value (S) of a creature is decided by its body shape and influences its ability to move, the Energy level (E) is a value that constantly changes as the creatures moves in its environment: it decreases by increased body movement. Figure 4 shows the correlation between Speed (S) and Energy (E).

Speed (S): depends on creatures body shape
                decides how fast the creature can move
Energy (E): $E = 1$ at birth
                Speed (S) of movement reduces E
                $E<0$ ..... creature becomes hungry
                $E>0$ ..... creature can mate

Fig. 4 Correlation between Energy and Speed level

Each movement a creature performs costs energy. When the energy level reaches a certain threshold the creature becomes hungry and needs to eat.

4.2. Metabolism

Food is provided by the RGB color values and the luminance values of each pixel within the video image. By random certain creatures have preferences for red (R), for green (G) or for blue (B) pixels as well as for their different luminance values. Within the initial "soup" of creatures a variety of different types of creatures with different food preferences is provided. When a creature has moved and its energy level (E) has dropped below the threshold of $E=0$ it becomes hungry: to reach its target the creature will move towards the pixel with the right type of color and luminance and metabolize its energy. This will help the creature to increase its internal energy level and, given that the level has risen to $E>0$, the creature will be ready to mate. Figure 5 shows this correlation between energy level, feeding and mating behavior.

Feeding: if $E<0$ .... creature wants to eat pixels
                it eats pixels with RGB and luminance values
                according to its genetic preferences

Mating: $E>0$ .... creature wants to mate, if successful,
                parents will exchange their genetic code
                -> a child creature can be born

Fig. 5 Correlation between feeding and mating behavior
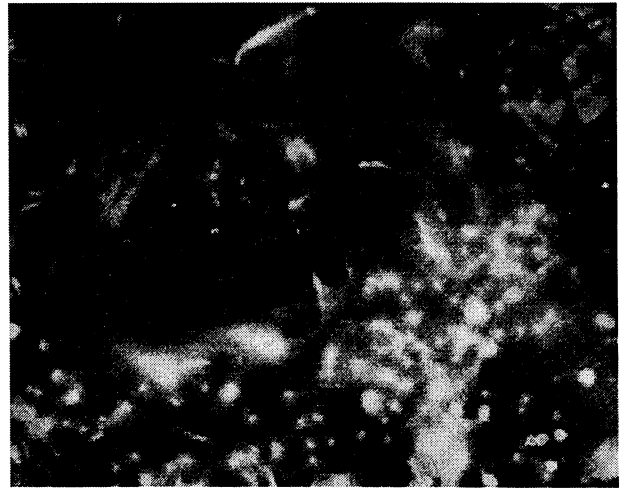
Fig. 6 Example of creatures eating certain pixels in the video image

Figure 6 shows how certain creatures have moved to certain parts of the video image to eat the particular pixels available at that location.

4.3. Reproduction

When two creatures have accumulated enough energy they can start to mate with each other and create an offspring creature. In this case, the offspring inherits the genetic code of the parent creatures; this is done through cross-over of the parents' codes and application of minimal mutation. Cross-over can take place at any part of the genetic string and the location and length of the cross-over is decided at random, however it is adapted to the length of the genetic string.

4.4. Evolution

The constant movement, feeding, mating and reproduction activities of the creatures result in a complex system of interactions with a selection for faster creatures. The user and her interaction decisions however will ultimately influence the creatures behavior and their possible evolution. By feeding and reproducing certain types of creatures it is anticipated that the users interaction and the internal behavior parameters of the creatures themselves can create a complex system that might display adaptive evolution with interactions between creatures and creatures as well as between users and creatures.

5. Conclusions

In our aim to create artworks that can be compared to living systems [13] PICO_SCAN represents a further attempt in the design of interaction that can link real life

data with artificial life data through human-machine interaction.
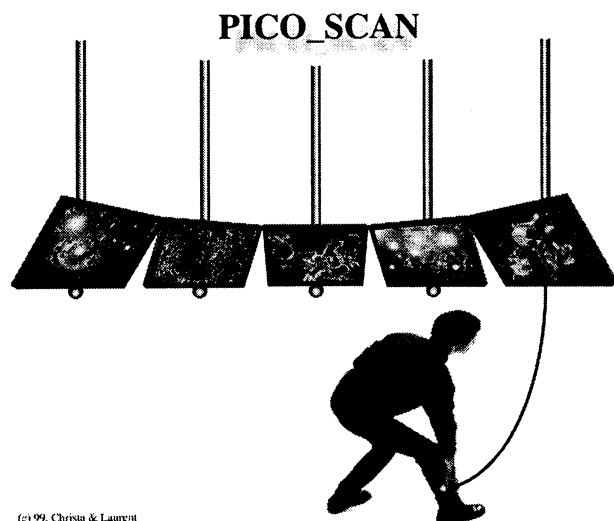
Fig.7 PICO_SCAN at the Martin Gropius Bau in Berlin April 2000.

Acknowledgement

The final exhibition of PICO_SCAN will take place at the Martin Gropius Bau in Berlin in April 2000. Figure 7 shows the final system set-up. The system was supported by the Berliner Festspiele GmbH and ATR MIC Labs Kyoto. We would like to especially thank Mr. Stephen Jones for his support on the hardware design.

References

[1] Sommerer C, Mignonneau L (1998) The application of artificial life to interactive computer installations. Artificial Life and Robotics Journal, Tokyo: Springer Verlag, Vol.2, No.4, pp. 151-156.

[2] Sommerer C, Mignonneau L (1993) Interactive Plant Growing. Ars Electronica 93 - Genetic Art Artificial Life. Vienna: PVS Verleger, pp. 408-414.

[3] Sommerer C, Mignonneau L (1997) Interacting with Artificial Life: A-Volve. Complexity Vol 2, No. 6, pp 13-21.

[4] Sommerer C, Mignonneau L (1995) Phototropy Oltre il villaggio globale - Beyond the Global Village. Milan: Electra Edition, p. 134.

[5] Sommerer C, Mignonneau L (1996) MIC Exploration Space. Siggraph'96 Visual Proceedings, New York: ACM Siggraph, p. 17.

[6] Sommerer C, Mignonneau L (1999), Life Spacies: a genetic text-to-form editor on the Internet, Proceedings AROB 4th'99, Beppu, Oita, pp. 73-77.

[7] Sims K (1994) Evolving 3D Morphology and Behavior by Competition. Artificial Life IV. Boston: MIT Press, pp. 28-39.

[8] Holland J H (1994) Echoing emergence: Objectives, rough definitions, and speculations for Echo--class models. Complexity: Metaphors, Models and Reality. Reading, MA: Addison--Wesley.

[9] Ray T (1991) An Approach to the Synthesis of Life. Artificial Life II. Redwood City, Calif.: Addison-Wesley, pp. 371-408.

[10] Jones S (2000) Sensing, Communication and Intentionality in Artificial Life. Proceedings AROB 5th'00, Oita

[11] Sommerer C, Mignonneau L (1999), Life Spacies II: from text to form on the Internet using language as genetic code. ICAT'99 Conference Proceedings, Tokyo

[12] Casti J (1999) The Art of Language. Complexity Journal, Vol.5, No.1. pp. 12-15.

[13] Sommerer C, Mignonneau L (1998), Art as a Living System. Art @ Science. Springer Verlag, Vienna New York, pp 148-161.

Proposal of a Function Discovery System Using Artificial Life Model and Its Applications

Seiichi SERIKAWA* and Teruo SHIMOMURA**

Dept. of Electrical Eng., Kyushu Institute of Technology, 1-1, Sensui-cho, Tobata-ku, Kitakyushu,
Fukuoka 804-8550 Japan, serikawa@elcs.kyutech.ac.jp*   simomura@elcs.kyutech.ac.jp**

Abstract   A function-discovery system named S-system using a bug type artificial life is proposed from an engineering point of view. Some known and unknown functions are discovered by this system. Many individuals that are called "bugs" exist in the system for this purpose. The bug is characterized with a chromosome consisting of functions, constants and variables. A tree structure is used for the expression of the chromosome. In the system, the existence of species with the same chromosome structure is regarded as homogeneity and with different chromosome structure is regarded as heterogeneity. The term sexual reproduction is used in case of a crossover between homogeneous bugs. The bug without homogeneous species gives birth to an exact copy by asexual reproduction. The concept of sexual and asexual reproduction is very effective in a function-search. The concept of movement is also important. The movement corresponds to a slight change of constants in the chromosome. This contributes to the improvement in local search ability. The validity of the system has been confirmed by using some observation data obeying known functions. As an example of engineering problem, design of optical lens is performed. Desirable lens shapes are automatically discovered.

Key words: artificial life, genetic programming, genetic algorithm, function discovery

1. Introduction

From the engineering point of view, study of artificial life (A-life) includes the Genetic algorithm (GA). GA is an effective method for the search of optimum values, and its fundamental principle consists of the selection and the combination based on fitness. Recently, Genetic programming (GP) has been proposed for applications except for optimum search[1]. The function search system using GP was first proposed by Koza[2]. This system expresses the function by a tree structure. Some unknown functions are discovered by the system. There are, however, the following disadvantages. (i) The schema tends to get destroyed by the crossover, (ii) the solution does not stabilize, and, (iii) the function length becomes extremely long or extremely short.

An improved system overcoming these disadvantages is proposed in this paper. We named the system as S-System because of the introduction of the concept of Sexual and asexual reproduction. In S-System, there exist many individuals called bugs. These bugs can evolve steadily without destroying the schema by sexual and asexual reproduction. The concept of movement is also important. The movement corresponds to a slight change of constants in the chromosome. This contributes to the improvement in local search ability. We have applied S-System effectively in some function search problems. As an example of engineering problem, design of optical lens is also performed.
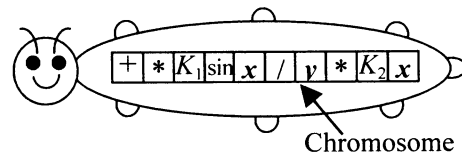
Fig. 1 The concept of the chromosome of a bug.

2. Model of a bug type of A-life

2.1 Structural expression of the bug's chromosome

In this system, there exist many individuals who have the chromosome shown in Fig. 1. For the sake of convention, we name each individual "bug". The chromosome of a bug consists of a set of function symbols; functions, constants and variables. For the expression of functions as chromosomes, a tree structure is adopted in the same way as GP[2]. The tree structure is identified by means of the S expression of the LISP language. For example, the function $K_1 \cdot \sin(x) + y/(K_2 \cdot x)$ is expressed in Fig. 2 (a) and (b) by the tree structure and S expression, respectively.

2.2 Meaning of function-discovery

First, we express all the experimental data D as follows.

$$D = \{\overline{d_1}, \overline{d_2}, \cdots, \overline{d_N}\}, \tag{1}$$

where N is the number of experiments. $\overline{d_i}$ denotes the i-th observation data and is given by Eq. (2).

(+ (* K_1 sin (x))

(/y(*K_2 x)))

(a) tree structure   (b) S expression of (a)
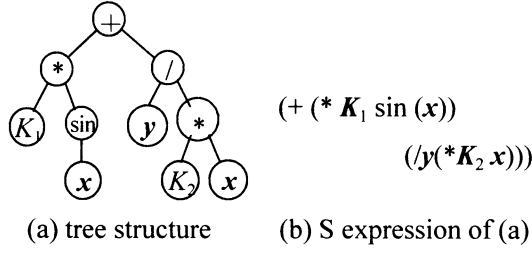
Fig. 2 Tree structure and S expression of the function $K_1 \cdot \sin(x) + y/(K_2 \cdot x)$.

$$\overline{d_i} = \left(In_{1,i}, In_{2,i}, \cdots, In_{t,i}, Out_i \right),$$ (2)

where $In_{1,i}, In_{2,i}, \cdots, In_{t,i}$ are the input data and Out_i, the output data. In the case of a high correlation between the input data and output data, the following equation will hold.

$$Out_i = f\left(In_{i,1}, In_{i,2}, \cdots, In_{i,t} \right),$$ (3)

where symbol f represents a certain function. It is our aim to discover an unknown function f using a bug type of A-life. When the chromosome of a bug p agrees with the function f, we assume that the bug p has discovered the function f.

2.3 Acquisition of internal energy

Before describing the algorithm of function-discovery, we define some equations. The fitness fit_p of bug p is defined as

$$fit_p = 1/(1 + \sigma_p),$$ (4)

where,

$$\sigma_p = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(Out_i - Bug_p(i) \right)^2},$$ (5)

where $Bug_p(i)$ denotes the value which is obtained by the substitution of the i-th observation data $In_{1,i}, In_{2,i}, \cdots, In_{t,i}$ into the chromosome of bug p. The square of Eq. (5) represents the mean square error between Out_i and $Bug_p(i)$. When $Bug_p(i)$ agrees with Out_i, then $\sigma_p = 0$ and $fit_p = 1$. In the case where the values of $Bug_p(i)$ and Out_i differ substantially, $\sigma_p \approx \infty$, so $fit_p \approx 0$. This means the range of fit_p is $0 \sim 1$. As fit_p approaches 1, the function $Bug_p(i)$ approaches the observation data Out_i. Regarding fitness fit_p as the internal energy of bug p, we can say that the calculation of the fitness corresponds to the acquisition of the internal energy by a bug. Therefore, the observation data is equivalent to "food" for the acquisition of internal energy.

2.4 Algorithm of function-discovery

Figure 3 is the flowchart of the algorithm of function-discovery making use of a bug type of A-life in this study. The flow is summarized as follows.

(1) Numerous bugs with an arbitrary function are
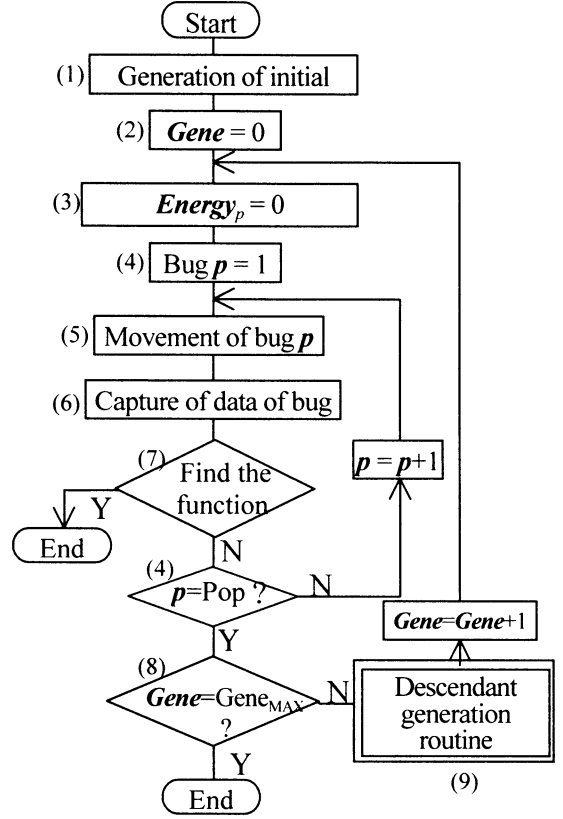
Fig. 3 The flowchart of the algorithm of function-discovery using the bug type of artificial life.

generated at random. The number **Pop** is selected from the numerous bugs in the order of high fitness.

(2) The generation **Gene** of the bug is set to 0.

(3) The value of the internal energies, $energy_p$, of all the bugs are initialized to 0.

(4) The procedures from (5) to (7) are repeated for all the bugs; the bug number ranges from 1 to **Pop**.

(5) The bug p moves. This means that the values of constants $\overline{K}$ in the chromosome change slightly. The value of $\overline{K}$ is replaced by $\overline{K} + \overline{dK}$, where $\overline{dK}$ is the small change in $\overline{K}$, where $\overline{K} = (K_1, K_2, \cdots, K_n)$, $\overline{dK} = (dK_1, dK_2, \cdots, dK_n)$ and n is the number of constants in the chromosome. This concept is based on Ref. (3).

(6) The bug p catches the observation data (i.e., fitness fit_p of bug p is calculated from the observation data).

(7) When the fit_p reaches the threshold fitness Fit_{TH}, the algorithm ends. This means that a bug has discovered the function f.

(8) The algorithm ends when the current generation **Gene** reaches the maximum Generation $Gene_{max}$.

(9) After the descendant-generation-routine is called, **Gene** is added to 1 and the algorithm returns to

procedure (3).

2.5 Descendant-generation-routine

The flowchart of descendant-generation routine is shown in Fig. 5, and can be described as follows.

(a) Based on the generation-gap[4], **Par** bugs are selected and they are passed down to the next generation. The elite strategy is adopted for the generation-gap.

(b) The bug number **p** is set to 0. By the repetition of the procedures from (c) to (g), **Pop - Par** bugs are generated.

(c) A bug is selected by tournament strategy.

(d) The selected bug is judged whether it has the ability to sexually reproduce. In the case that the selected bug has the ability of sexual reproduction, procedure (e) is performed. In the other case, procedure (f) is carried out.

(e) The bug finds its partner, and they produce two children by crossover. Jump to procedure (g).

(f) Two children are produced by asexual reproduction.

(g) A part of the chromosome is changed by mutation at the rate of R_{mut}.

In this way, the descendants of the number of **Pop** are generated.

2.6 Sexual reproduction and asexual reproduction

The concept of sexual and asexual reproductions is introduced in this study. This is important for an effective function-search. The existence of species with the same chromosome structure is regarded as homogeneity and with different chromosome structure is regarded as heterogeneity. The term sexual reproduction is used in case of a crossover between homogeneous bugs. Only the constants change by sexual reproduction. The bug without homogeneous species gives birth to an exact copy by asexual reproduction. If this concept is not introduced, a bug which does not belong to any other species cannot evolve at the next generation.

GP blindly applies a crossover. As a result, the schema (profitable partial tree) tends to be broken. In S-System, the tree structure does not change due to a crossover because of sexual/asexual reproduction. The bug evolves without breakup of schema.

However, the crossover of the S-system lacks the variety of functions. For the variety, we introduce a mutation and a limitation of the number of homogeneous bugs.

3. Application to function-discovery

For confirming the validity of the S-system, we attempted to search for the functions using the
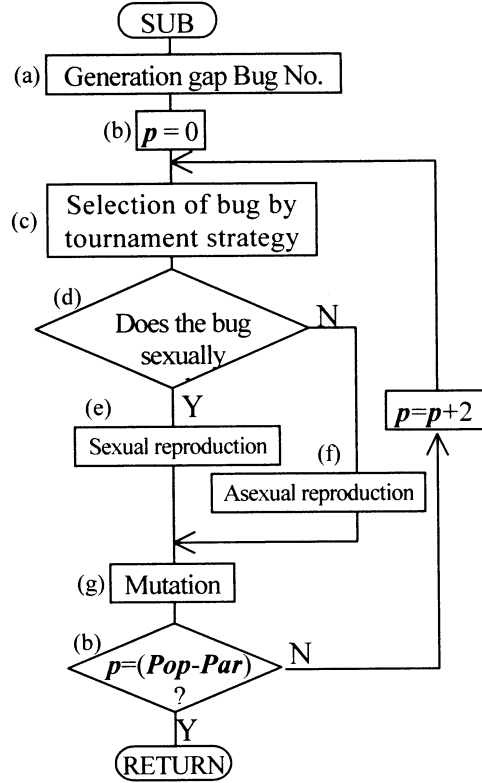
Fig. 4 The flowchart of descendant-generation routine.

observation data which obey Kepler's law, Ohm's law and Snell's law. The function fitting each law has been successfully discovered. In these results, the search for Kepler's law is mentioned here.

Let the major axis-length of a plane orbit be **D** (normalized by the earth's length) and a period of revolution be **P**, then Kepler's law is expressed

$$\frac{D^3}{P^2} = \text{Const.} \tag{6}$$

To rearrange the expression, $P = K_1 \cdot D^{K_2}$, where K_1 and K_2 are constants.

The observation data (D, P) corresponds to the data $(In_{1,i}, Out_i)$ in Eq. (2). We included the term for power in the function set in addition to $+$, $-$, $\times$, because almost all the laws, which were discovered in the earlier part of twenty century, included the terms for power[5]. Since Kepler's law is easily found when the term for $\sqrt{}$ is included, we omitted the term.

Figure 5 shows the results of the function search. The fitness in the figure represents an average of 30 experiments and corresponds to the fitness of the bug that has produced the highest fitness. It is seen that the fitness of the final generation of S-System is much higher than that of GP. In addition, the mean function length of S-System is far shorter than that of GP.

Although the GP procedure was attempted 30
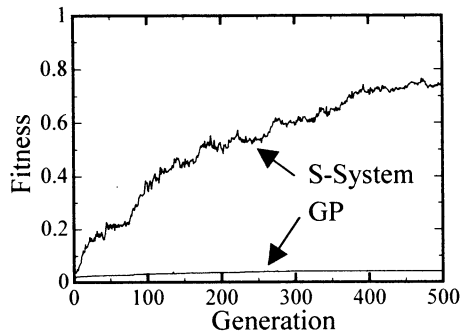
Fig. 5 The average of 30 experiments showing the relation between the generation and the fitness of the bug with the highest fitness.

Table 1 Comparison between the values of m_{MDL} and σ_{MDL} of GP and those of the S-system for Kepler's law (m_{MDL}: mean value of MDL, σ_{MDL}: standard deviation of MDL, trial: 30 times).

	GP	S-system
m_{MDL}	11.2	-8.7
σ_{MDL}	2.6	2.6

times, the function in agreement with Kepler's law was not obtained.

In S-System, the fitness increases steadily without the breakup of schema because of sexual and asexual reproduction. The values of constants slightly change by the concept of movement. By these strategies, S-System is found to be excellent in the local search ability compared to GP.

For the quantitative comparison of GP and the S-system, we calculate the values of *Minimum Description Length* (*MDL*)[5] 30 times. The mean value m_{MDL} and standard deviation σ_{MDL} are shown in Table 1. Obviously, the value of m_{MDL} using the S-system is lower than that using GP although the values of σ_{MDL} between them are similar.

4. Application to Lens Design

Since the shape of lens can be represented by a function, we tried to design a lens using S-System. As far as lens design is concerned, we defined the fitness so that it increases when the light concentrates at a point and the thickness of the lens is thin. Figure 6 shows the process of the evolution of the lens. A globular lens cannot evolve by limiting the thickness. As shown in the figure, the shape approaches to that of a Fresnel lens and light concentrates as the generation advances.

Besides this, various lenses were designed by changing the condition of the fitness.

| The first generation | The 10th generation | The 100th generation |
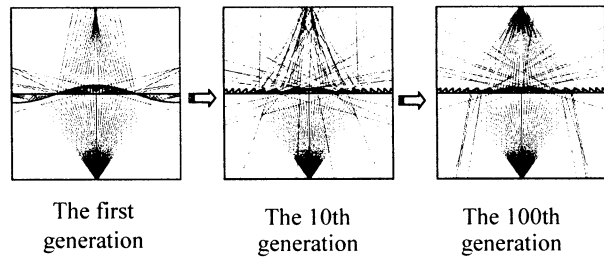
Fig. 6 The process of the evolution of the lens.

5. Conclusions

The effectiveness of artificial life is investigated from an engineering point of view. The function-search system (named S-system) using a bug type of artificial life is proposed in this study. The chromosome of a bug consists of functions, constants and variables. The concept of sexual and asexual reproduction and the movement of bugs are newly introduced. These contribute to the improvement in local search ability. As the generation proceeds, the bugs with the functions in agreement with the observation data survive selectively, and determine the function corresponding to the given data. S-System is found to have the following advantages: (i) the schema is not destroyed by the crossover, (ii) the solution stabilizes, and, (iii) the function length is relatively short. As an example of engineering problem, the design of lens is attempted. The desirable lens shape is discovered.

References

[1] J. Koza, *Genetic Programming, On the Programming of Computers by Means of Natural Selection, MIT Press*, (1992).

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

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

[4] Goldberg, D. and Deb, K., "A Comparative Analysis of Selection Schemes user in Genetic Algorithm", *Foundations of Genetic Algorithms*, Rawlins, G.J.E. (ed.) Morgan Kaufmann (1991).

[5] Iba, H., *Basis of Generic Algorithm*, Ohm, p.102, p.138, p.153 (1994).

Prediction of moving object position in visual perception system of artificial life

Wen Nian [†] & Kozo Okazaki[†] & Shinichi Tamura[‡]

[†]FUKUI UNIVERSITY

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

nian@vision.fuee.fukui-u.ac.jp

[‡]OSAKA UNIVERSITY

D11,2-2 Yamadaoka,Suita,Osaka 565-0871,Japan

tamuras@image.med.osaka-u.ac.jp

Abstract

In this paper, we generate creatures having stereo perception ability composed of a neural network. Stereo perception ability is a basis for the spatial perception. This spatial perception ability is also a basis of predicting future target position which we can regard one of the spatial reasoning functions of the creatures. We present two models of binocular stereopsis. The search spaces for stereo perception and prediction by GA are enormously vast and are difficult to evolve well the creature. By using the idea of mammalian evolution, we divide them to children and adult groups and promote to evolve space perception and prediction abilities.

keywords: evolution, vision system, artificial life, genetic algorithm, space perception, prediction.

1 Introduction

Pattern recognition is one of the important tasks in intelligent information processing. Then it may be valuable to learn from the function of creatures to improve this task. However, there is questions whether the vision system of creature is the optimum, or whether another methods can exist. Therefore, to check the necessity of vision system of a creature, we have been studying the evolution of visual system on a computer assuming that the visual organ of a creature is composed of neural network with arbitrary structure. The structure of the system is determined and evolved by a genetic algorithm(GA).

In this paper, we use a binocular stereopsis model (Fig.1), and derive the stereo perception ability of the creature, which we may be able to extend to a neural network able to predict future target position which we can regard as a spatial reasoning function.

2 Binocular stereopsis and 2D position recognition

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

2.1 Modeling of binocular stereopsis

Generally, both a point $P(x, d)$ of space and perspective mapping point on an imaging plane are real number. Therefore, the searching space of it become enormours and need to approximate to integer, where the error causes perception and prediction error. We propose two models to cope with them.

(1) model 1
A binocular stereopsis **model 1** is shown in Fig.1. The relation between the target position (x, d) in 2D space and the corresponding target image positions in left and right eyes, X_R and X_L are given by Fig.1 and Eq.(1) . The position recognition and prediction are done only for square latticepoints of x and d.

$$\begin{cases} X_L = K_M * f * x/d \\ X_R = K_M * (L * (d - f) + f * x)/d \end{cases} \tag{1}$$

where K_M is a function makes all space positions not overlap as rounded integer value as far as possible. The point O is set as the origin of X_L and

X_R to make the value of X_L and X_R non-negative number.
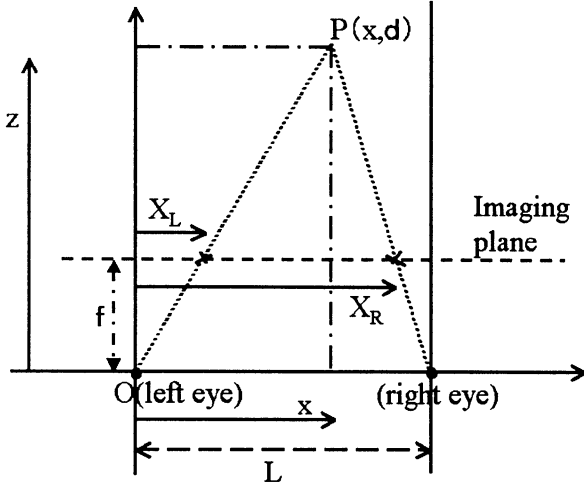
Figure 1: Model 1 of binocular stereopsis

(2) model 2

A binocular stereopsis **model 2** is shown in Fig.2. The origins of left and right eye are "O_L" and "O_R", respectively. We obtain Eq.(2) from Fig.2. The position recognition and prediction are done only for crossing points.

$$\begin{cases} X_L = f * x/d \\ X_R = f * (x - L)/d \end{cases} \quad (2)$$
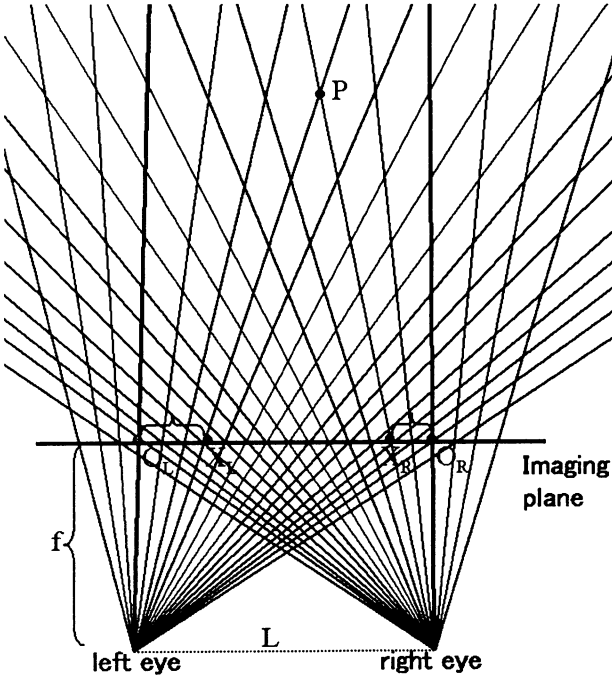
Figure 2: Model 2 of binocular stereopsis

2.2 Stereo perception

The neural network that can output simply the target depth from two one-dimensional images corresponding to epipolar lines for simplicity will be constructed. Here, output plane is one-dimensional array as shown in Fig.(3). A two-dimensional position map with horizontal position axis and depth axis will be possible as the output.

(1) model 1

The visual depth d and horizontal position x are obtained by Eq.(3).

$$\begin{cases} d = K_M * L * f/(K_M * L + X_L - X_R) \\ x = X_L * L/(K_M * L + X_L - X_R) \end{cases} \quad (3)$$

Equation (3) indicate that the horizintal position x and depth position d are calculated separately.

(2) model 2

The visual depth d and horizontal position x are obtained by Eq.(4).

$$\begin{cases} d = L * f/(X_L - X_R) \\ x = X_L * L/(X_L - X_R) \end{cases} \quad (4)$$

Equation (4) indicate that the horizintal position x and depth position d are calculated separately. Network is composed by mutually connected neural network(NN).
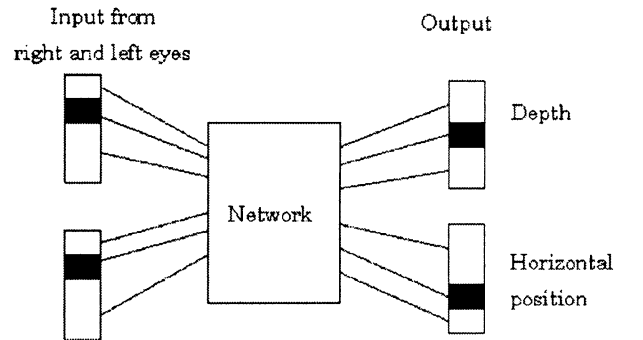
Figure 3: Stereo perception networks in depth and horizontal position

2.3 Prediction of moving object

The neural network that can predict the target position $P_3(x_3, d_3)$ in the next time from two one-dimension images $(P_1(x_1, d_1), P_2(x_2, d_2))$ which express target directions at the two consecutive times

will be constructed. The structure is shown in Fig.4, where the output is two one-dimensional image.

The neural network that can output depth and horizontal position of the predicted target position from four one-dimensional images in consecutive times can be constructed. Intermediate output is four one-dimensional arrays corresponding to depth and horizontal position at each time. As an alternative expression, we can obtain two two-dimensional recognition maps as shown in Fig.4. Finally,they will be converted to a position prediction map at the next time.

The prediction of moving object is obtained by Eq.(5)

$$\begin{cases} x_3 = x_2 + (x_2 - x_1) \\ d_3 = d_2 + (d_2 - d_1) \end{cases} \qquad (5)$$

This structure is also applicable to generate an absolute environmental map including positions of target and the creature itself.
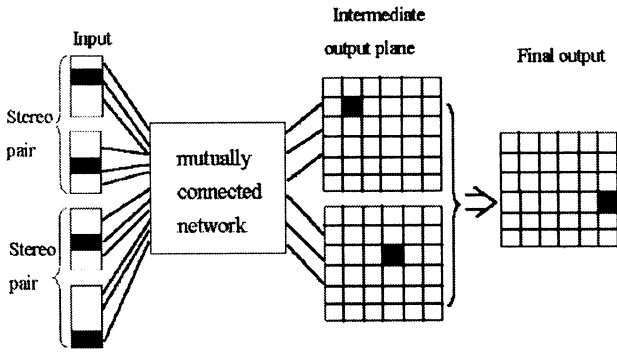
Figure 4: Generation of recognition map for 2-D moving target position prediction

3 Coding of GA and promotion of evolution

3.1 Individual genotype

We must set form of creature's gene when we apply GA. Here,we regard the configuration shown in Fig.3 as creature's individual. So we assign the connection weights among all pairs of units and the thresholds of processing units to the gene code. Thus the visual system of virtual creature composed of the neural network is represented by the gene. Practically, the network is realized separately in depth and horizontal position calculation and are perceived in the same structures and genotypes.

This is shown in Fig.5 where we implement a NN with muturally connected (processing unit;PU)NN. In Fig.5,M ,N and K are the neuron number of input, PU and output of PU,respectively.

For example, In this case for space perception using model 1, we set M ,N,K as 26,6 and 5, the searching space formaly becomes $5^{26\times6+(6+5)/2+6\times5} \times 9^{6+5}$.
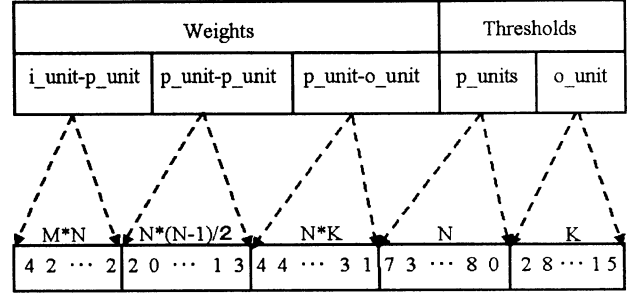
p_unit ,i_unit and o_unit mean processing unit,

input unit, and output unit,respectively

Figure 5: Structure of creature genotype for space perception (Fig.4)

3.2 Generation replacing rule

We set the condition that the creature will survive if it is correct to detect the distance (or horizontal position) for the input images, and it will die if it is wrong. In this case, however, all of the creatures will die out soon , even though we set the initial number of creatures enough large as 1000. The search spaces for stereo perception and prediction by GA are enormous and are difficult to evolve the creature well. By using the idea of mammalian evolution, we divide them to children and adult groups and promote to evolve space perception and prediction. Therefore, we divide the creatures into two groups to improve survival and evolution rate; a children's group and an adult group. The number of creatures are both (50*G), here G is the upper limit age of children group. The number of creatures generated randomly at every generation from the first to tenth was set 50. We adopt the criterion function (correct detection number /ages) as shown in Eq.(6).

Here, we set one generation as one year and the creature's correct detection number is added 1 and one new creature is generated by crossing with another creature in the adult group randomly for every one generation when the detection is correct. The new creature is shifted to the children's group. The creatures with ages lower than (G+1) belong

the children's group, In this group, all of the creatures are calculated its criterion, and no operation of death-life and multiplication is applied. When the creature's age is over G, they are shifted to the adult group and are applied the operation of death-life and multiplication as shown in Fig.6.

From (G+1)'th, new creatures generated by creatures of adult's group are shifted to the children's group. In the adult group, the creatures are arranged in the order of priority (criterion function). If the number of creatures becomes 50*G, the individuals with the lowest priority-order are removed.

$$priority\_order = \frac{correct\_detection\_number}{ages} \quad (6)$$
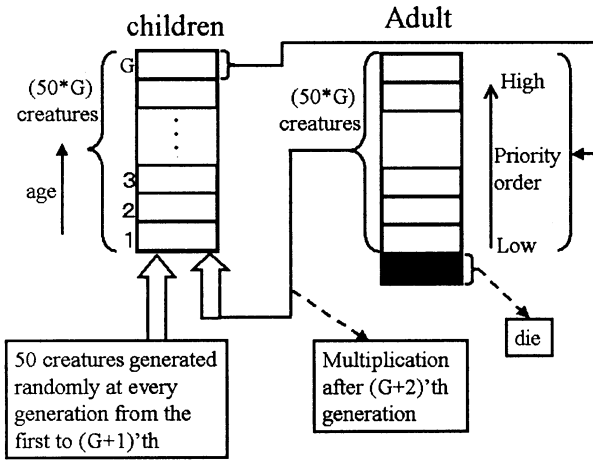
Figure 6: Profile of children and adult group and creature's multiplication

4 Experiment

When the target is far from the eyes, the accuracy of the distance perception becomes low. Therefore, we set the distance is classified only to 3,5,8,13 with $f = 2, L = 4$. That is, the target distance d is calculated in the network from X_L and X_R, and output from (fire) one of the four neurons corresponding to distance 3,5,8,or 13. We assume the middle layer is composed of a mutually connected network. Each connection between two units is bidirectional with the same weight. Calculation in mutually connected NN should be done by solving the variables of inputs, those of processing units and outputs. However, there include nonlinear elements in NN. So far, the outputs are determined by repeating the calculation until the outputs become stable. We set the number of repetition as 20. The

output (or state) of an unit being excited or suppressed is determined by a threshold value. The number of processing units is 6, and the number of output units is 5. The weight's value is set as one in the set [-3,-2,-1,0,1,2,3]. Here, the threshold of output units is set as 0,and the threshold of processing units is set [-4,-3,-2,-1,0,1,2,3,4]. The weights of connection between the units are determined by GA. Here, the crossover was done by uniform rate of 0.75. The mutation rate was taken one from the set [0.15,0.11,0.08,0.06,0.04] for every 1000 generations cyclically. Thus the neural network of the vision system is evolved by GA.

4.1 space perception

(1) model 1

The relation between the target position (x, d) in 2D space and the corresponding target image positions in left and right eyes, X_R and X_L are given by Fig.1 and Eq.(3). $f = 2, L = 4$.

(2) model 2

Calculated positions of X_L and X_R take only the points on imaging plane which cause no real-integer tranform errors as shown in Fig.2. $f = 11, L = 6$.

4.2 Prediction of moving object

The searching space in predition of moving object become far more enormous than that of space perception. The searching space formaly becomes $5^{52 \times 8 + 8 \times 7/2 + 8 \times 5} \times 9^{8+5}$ $(M = 52, N = 8, K = 5)$. We set the limitations of moving directions to 2 cases ((1)horizontal and (2)vertical).

5 Results and conclusive remarks

Table 1 shows the number of creatures correctly outputting the horizontal and vertical positions (x, d) of the target from imaging plane data among top 50 creatures. Best recognition rate in the table represents the recognition rate of the most superior individual for the above 28(7 × 4) tasks of the depth perception. The bottom row shows the average recognition rate of the top 50 creatures for the 28 tasks. Here (a) is for model 1 and (b) is fpr model 2.

Table 2 shows the number of creatures correctly outputting the horizontal and vertical positions (x, d) of the target from imaging plane data among top 50 creatures. Best recognition rate in the table represents the recognition rate of the most superior in-

dividual for the above 28 (model 1) or 36 (model 2) tasks of the depth perception. The bottom row shows the average recognition rate of the top 50 creatures for the tasks. We will come up to implement the second stage of two dimensional spatial recognition map. The third stage of our task is to design the neural network with which human being objectify oneself and model one's action in brain. There, the positions of himself and the target might be plotted on a map fixed on the earth(absolute map).

We will also apply the recognition map to carry out such activities that several targets emerge in time series individually or simultaneously. These situations correspond to the emergence of plural baits or enemies.

We are considering

(1) generating the absolute map based on the relative recognition map expressing the view from himself and his present action.

(2) drawing up the action plan based on these situations.

References

[1] W. Nian, K. Okazaki, and S. Tamura, "Evolution of Vision System by Genetic Algorithm", Proceedings of International Symposium on Artificial Life and Robotics (AROB '97), pp.9-13,1997.

[2] Hopfield, J.J., " Neurons with graded response have collective computational properties like those of two-state neurons", Proceedings of the National Academy of Sciences, 81, pp.3088-3092, 1984.

[3] Yoh-Han Pao, "Adaptive Pattern Recognition and Neural Networks", Addison Wesley, 1989.

[4] J.H.Holland, "Adaptation in Natural and Artificial Systems", The Univ. Michigan Press (1975), MIT Press (1992).

[5] T.Nagao, T.Agui and H.Nagahashi, "Structural evolution of neural networks having arbitrary connections by a genetic method", IEICE Trans. INF. & SYST. Vol. E76-D, No.6, pp.689-697. 1993.

Table 1: Spacial perception (space:7 × 4)

(a) model 1

	x					d				
	generation					generation				
	100	2000	3000	4000	5000	100	4000	10000	30000	45000
best recognition rate	.393	.50	.679	.857	1.0	.357	.571	.929	.964	1.0
average recognition rate	.236	.431	.631	.777	.966	.271	.529	.926	.964	.967

(b) model 2

	x					d				
	generation					generation				
	100	3000	5000	10000	45000	100	2000	5000	8000	9000
best recognition rate	.308	.462	.731	.808	1.0	.308	.423	.731	.808	1.0
average recognition rate	.203	.429	.723	.798	.970	.308	.385	.671	.771	.988

Table 2: Prediction of moving object (space:7 × 4)

(a) Depth direction (model 1)

	x at time t_3					d at time t_3				
	generation					generation				
	100	2000	3000	4000	5000	100	3000	10000	30000	65000
best recognition rate	.285	.571	.750	.929	1.0	.357	.536	.750	.893	1.0
average recognition rate	.250	.516	.710	.881	1.0	.279	.491	.744	.890	.967

(b) Horizontal direction (model 2)

	x at time t_3					d at time t_3				
	generation					generation				
	100	5000	20000	50000	80000	100	2000	4000	5000	9500
best recognition rate	.222	.528	.667	.833	.889	.333	.528	.667	.927	1.0
average recognition rate	.171	.486	.568	.822	.854	.327	.480	.643	.911	.996

Application of Stereovision Neural Network
to Continuous Speech Recognition

Tomoyuki Ichiki, Tetsuro Kitazoe

Department of Computer Science and Systems Engineering
Faculty of Engineering, Miyazaki University
1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192 Japan
e-mail: ichiki@cs.miyazaki.-u.ac.jp ,kitazoe@cs.miyazaki-u.ac.jp

ABSTRACT

The two or three layered networks 2LNN, 3LNN which originate from stereovision neural network are applied to speech recognition. To accommodate sequential data flow, we consider a window to which new acoustic data enter and from which final neural activities are output. Inside the window recurrent neural network develops neural activity toward a stable point. The process is called Winner-Take-All(WTA) with cooperation and competition. The resulting neural activities clearly showed recognition of a continuous speech of a word. The string of phonemes obtained is compared with reference words by using DP matching.

1. INTRODUCTION

Since we recognize speech through neural network in the brain, many works on this line have been conducted for speech recognition. Though probabilistic acoustic models represented by Hidden Markov Model (HMM) has been met widely used so recognizers today, it has been long standing go to let machine enable human abilities of speech recognition in the brain. Various kind of neural networks have been proposed for speech recognition such as multilayer perceptrons (MLP)[1,2], time delayed neural network (TDNN)[3], hidden control neural network (HCMM)[4], hybrid system combining HMM and MLP (HMM/MLP)[1] and fully recurrent neural network (FRNN)[5,6], notable things are that these models use more or less learning algorithms of back propagation of error and need many parameters to be adjusted.

In the previous works, we employed a new approach to the problem by applying stereo vision neural network to hearing system[7,8,9,10,11]. The neural networks are two or three layered (2LNN, 3LNN) and the parameters in the equations are fixed and not changed at any time. We assumed that the feature parameters characteristic of each phoneme were stored or memorized in the brain in the form of probability density functions and that this processes were considered as learning processes. The neural network equation employed from visual system process the similarities between the characteristic phonetic features stored in our memory and the input acoustic data from our ears. The resulting phoneme recognition rate was fairly good, resulting 7-9% higher than HMM model. In the present paper we are going to give an algorithms for the continuous speech recognition. The major problem in this case are how to introduce an algorithms to that the real time acoustic data flows and how to employ the neural network to process the data flows, giving the recognition of continuous speech.

2. APPLICATION OF NEURAL NETWORK TO SPEECH RECOGNITION

The speech(phoneme) recognition system using stereo vision neural net equations is divided into four main processes;

(1) A number of training speech data are classified and parameterized into sequences of feature vectors for each phonemes. The feature vectors are used to form standard Gaussian PDFs which are supposed to be memorized in our brain for each phoneme.

(2) An input phonemes are referred to these memorized phoneme data and a similarity measure is obtained by comparing the input phoneme data with the memorized PDF of each phoneme.

(3) Suppose that there is a neuron activity ξ_a^u according to the similarity measure λ_a^u to a certain phoneme /a/ at the frame number u.

(4) The stereo vision neural net equations are performed to make an activity ξ_a^u move toward a stable point after the equations receive the similarity measure as an input and a recognition results are achieved when it reaches to a stable state.

The memorized standard acoustic models for each phonemes are expressed in terms of Gaussian PDF for input o.

$$N(o;\mu_a,\Sigma_a) = \frac{1}{\sqrt{(2\pi)^n|\Sigma_a|}} e^{-\frac{1}{2}(o-\mu_a)^t \Sigma^{-1}(o-\mu_a)} \quad (1)$$

where μ_a is a mean value and Σ_a is covariance matrix, of feature vectors for training data of a phoneme /a/. The normalized similarity λ_a^u of input data O_u at u-th frame to a certain phoneme /a/ is defined as

$$\lambda_u^a = \frac{\log N(o_u;\mu_a,\Sigma_a) - <\log N>}{<\log N>} \qquad (2)$$

where <logN> means an average over phonemes at the same frame.

3. THREE LAYERED NEURAL NET EQUATIONS

Since 2LNN has a similar property with 3LNN, we discuss have 3LNN which is given as

$$\tau_1 \dot{\xi}_u^a(t) = -\xi_u^a(t) + f(\beta_u^a) \qquad (3)$$

$$\tau_2 \dot{\alpha}_u^a = -\alpha_u^a + A\lambda_u^a - B\sum_{a'\neq a} g(\xi_u^{a'}(t)) + D\sum_{u'=n-l1}^{u+l2} g(\xi_u^a(t)) \qquad (4)$$

$$\tau_3 \dot{\beta}_u^a = -\beta_u^a + g(\alpha_u^a) + g(\xi_u^a) \qquad (5)$$

f(x) is a well known as sigmoid function and g(u) is a function given by

$$f(x) = (\tanh(w(x-h)) + 1)/2 \qquad (6)$$

$$g(u) = u^+ = (u + |u|)/2 \qquad (7)$$

where A,B,D,w,h are positive constants which are to be chosen appropriately.

Figure 1 shows three layered structure of the stereo vision neural network.
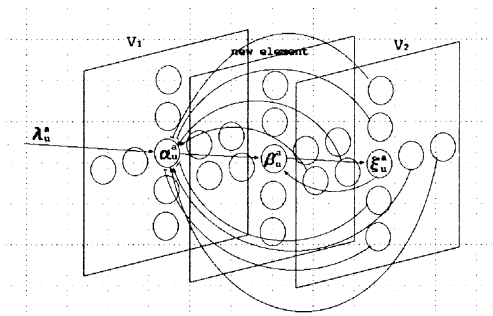
FIGURE 1. Three layered neural network(3LNN)

To understand the qualitative feature of the equations, consider equilibrium solution $\dot{\alpha}_u^a = \dot{\beta}_u^a = \dot{\xi}_u^a = 0$.
Equa-tions (3)(4)(5) are written as

$$\xi_u^a(t) = f(g(\alpha_u^a) + g(\xi_u^a)) \qquad (8)$$

As illustrated in figure 2, the solutions are given by the intersection of the two curves. We obtain following two conclusions from this figure.
(1)For large α, ξ has high value(approximately 1) while it has low value(approximately 0) for small α.
(2) The solution ξ has different paths according to whether α is increased or decreased (hysteresis).
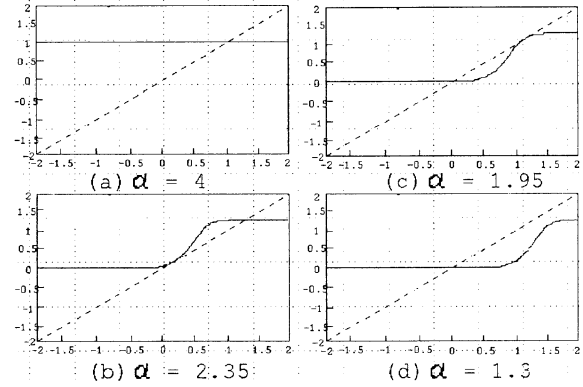
FIGURE 2. Curves of y = ξ (dotted lines) and
y=f(g(α)+g(ξ)) (solid lines)

4. CONTINUOUS SPEECH RECOGNITION

For continuous speech, it is considered that input data λ_u^a are fed to the neural network (3), (4), (5) and the activities α, β, ξ in the network develop their value toward a stable point. We judge as that a phoneme /a/ is recognized at a flame u if ξ_u^a goes to 1, while it is not recognized if ξ_u^a goes to 0. In the present study, to save calculation time, we assume $\tau_2, \tau_3 \ll \tau_1 = 1$ so that equation (3), (4), (5) are simply written as

$$\dot{\xi}_u^a(t) = -\xi_u^a(t) + f(g(\alpha_u^a) + g(\xi_u^a)) \qquad (9)$$

$$\alpha_u^a = A\lambda_u^a - B\sum_{a'\neq a} g(\xi_u^{a'}(t)) + D\sum_{u'=n-l}^{u+l} g(\xi_u^a(t)) \qquad (10)$$

Notice that equation (9), (10) give the same solution as (3), (4), (5) if the stable solution is unique. Differential equations (9), (10) are described as a loop of N steps of a loop in numerical calculation divided by discrete time Δt. Thus, ξ develops as

$\xi(t)$, $\xi(t + \Delta t)$, $\cdots$ to $\xi(t + N\Delta t)$.

To treat data flow λ_u^a sequentially, we consider that the data enter into a window with L frames and the neural network (9), (10) processes the L frame data for N steps. Then, the data are passed one frame forward through the window, where the initial values of ξ are set to zero before entering to the window. The procedure is stated more in detail as follows; When input data λ_{u+L-1}^a , λ_{u+L-2}^a , ..., λ_u^a are entered inside the window, equations (9), (10) develop activities $\xi(t)$ until it arrives at $\xi_{u'}^a(t + N\Delta t)$ for $u' = u, u+1, ..., u+L-1$. Then, new data λ_{u+L}^a enter into the window from the left and the old data λ_u^a get out from the window. At the same time whole $\xi_{u'}^a$ are shifted to $\xi_{u'-1}^a$, setting the initial values of $\xi_{u'-1}^a$ for the next N step loop calculations. At this time we have new comers ξ_{u+L}^a entering into the window and the final ones ξ_u^a output from the window. We eventually judge if phoneme /a/ at u-th frame is recognized or not according to whether the final ξ is close to 1 or 0, respectively. A sequence of the same processes continues until whole input data go through the window completely and whole values of final ξ are obtained.

5. EXPERIMENTAL RESULTS

We extracted a total 24 labeled phonemes from ATR Japanese speech database composed of 4000 words spoken by 10 male speakers and from ASJ speech database of 500 sentences by 6 male speakers to make Gaussian PDFs for each phoneme. For recognition test which is independent, we take 42 words of the training data. The experimental conditions are as follows

Sampling rate	16kHz,16bit
Pre-emphasis	0.97
Window function	16ms Hamming window
Frame period	5ms
Feature parameters	10-order MFCC
	+10-order Δ MFCC

In figure 3 the typical result for a word pronounced

/i/ /k/ /u/ /j/ /i/ is shown for the best two ξ_u^a output from the window. In the figure the best ξ are read sequentially as /i/ /h/ /t/ /k/ /u/ /j/ /i/. It is noticed that /y/ has rather high values following to /i/, because /y/ resembles to /i/. /h/ and /t/ do not have correspondence in the reference word and may be entered as context dependent effects between /y/ and /k/.

To get the recognition rate, a sequence of phonemes obtained from the window are shortened into a word. A phoneme is omitted if it appears isolated less than continuous three frames. Thus, AAABBCCC... is written as AC in short. The example in figure 3 is written as /i/ /h/ /t/ /k/ /u/ /j/ /i/.
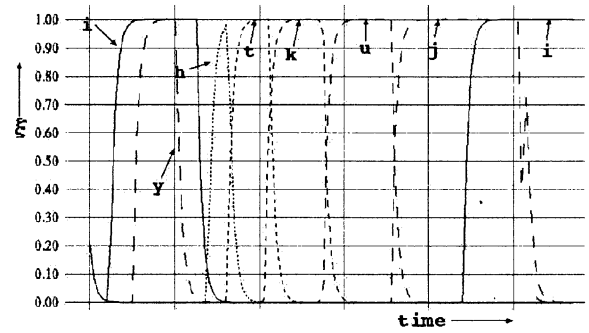
FIGURE 3. Best two of ξ are selected and plotted against frames(time). Here parameters are set as window size L=10 Δ t=0.01,N=100, A=6.0,B=6.5,D=1.5,w=2.5,h=0.5,l1=6,l2=0

After this preparation, we make DP matching where we define distance table between phonemes, by making several groups of similar phonemes. The resulting DP matching is shown in figure 4, comparing ihtkuji with ikuji. A lattice point shows a distance between phonemes. In the example, we set d(i,i)=d(u,u)=d(j,j)=d(k,k)=0, d(k,h)=d(k,t)=0.4 and the total distance D becomes D= 0.8.

Experiment is performed in this way and 38 words are correctly recognized among 42 independent words, giving 90% recognition rate.
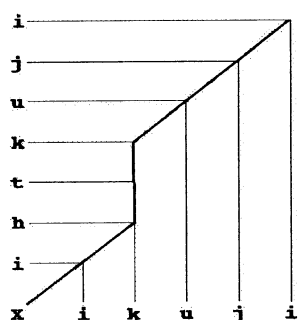
Figure4. DP matching between test word ihtkuji and reference word ikuji. X is the damy point allowing possibility of excess phonemes at the beginning.

7. CONCLUSIONS AND DISCUSSIONS

The two or three layered networks 2LNN, 3LNN which originate from stereovision neural network are applied to speech recognition. To accommodate sequential data flow, we consider a window to which new acoustic data enter and from which final neural activities are output. Inside the window recurrent neural network develops neural activity toward a stable point. The process is called Winner-Take-All (WTA) with cooperation and competition. The typical dynamics is shown in figure 5 where cooperation works in the neighboring frames at the phoneme, while competition does with other phonemes at the same phoneme. The resulting neural activities clearly showed recognition of a continuous speech of a word. The string of phonemes obtained is compared with reference words by using DP matching. The recognition results are 90% though we need further study with more input data. The nice feature of our model is that it does not have many parameters to be adjusted and the algorithm for recognition is very simple.
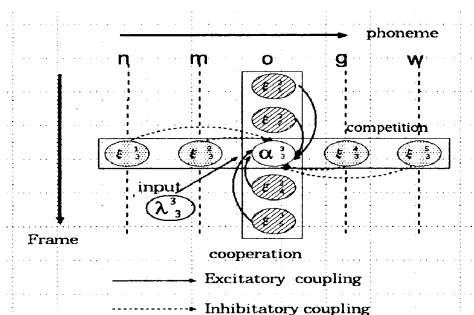
FIGURE 5. Dynamic al process of neural activities with cooperation and competition

Reference

[1] Bourlard, C.J.Wellekens. "Link between Markov Models and Multi-layer Perceptoron" IEEE Trans. Patt. Anal. Machine Intell., Vol.12, pp.1167-1178, 1990

[2] Hung, A.Kuh. "A Combined Self-Organizing Feature Map and Multilayer Perceptron for Isolated Word Recognition" IEEE Trans. on Signal Processing, Vol.40,pp.2651-2657, 1992

[3] J.Lang, A.Waibel, G.E.Hinton. "A Time-Delay Neural Network Architecture for Isolated Word Recognition" Artificial Neural Networks, Paradigms, Applications and Hardware, 1992

[4] Martinelli. "Hidden Control Neural Network" IEEE Trans. on Circuits and Systems, Analog and Signal Processing 41(3):245-247,1994

[5] T.Robinson(1992) Recurrent Nets for Phone Probability Estimation. Proceedings of the ARPA Continuous Speech Recognition Workshop, Stanford, Sept.

[6] Williams,R.J.,Zipser.D.(1990)Gradient based Learning Algorithms for Recurrent Connectionist Networks. Tech. Rep. NU=CCS-90-9,Northeastern University, College of Computer Science, Boston

[7] T.Kitazoe, J.Tomiyama, Y.Yoshitomi, and T.Shii "Sequential Stereoscopic Vision and Hysteresis" Proc. of Fifth Int.Conf. on Neural Information Processing, pp. 391-396, 1998

[8] T.Kitazoe,S-I.Kim,T.Ichiki. Speech recognition using Stereovision Neural Network Model. Fourth International Symposium on Artificial Life and Robotics, pp.576-579,Vol2, January, Beppu, Oita, Japan, 1999.

[9] Acoustic Speech Recognition Model by Neural Net Equation with Competition and Cooperation(Tetsuro Kitazoe,Tomoyuki Ichiki,Sung-Ill-Kim) ICSLP'98(The 5th International Conference on spoken Language Processing,Vol 7.pp3281-3284,30th November-4th December, Sydney,Australia,1998

[10]T.Kitazoe,S-I.Kim,T.ichiki,M.Funamori.Ac oustic Models in Speech Recognition by Stereo Vision Neural Nets. International Conference on Speech Processing,pp81-86,Vol1,August,Seoul,Ko rea,1999

[11] T.Kitazoe, S-I.Kim, T.Ichiki, M.Funamori. Acoustic Speech Recognition by Two and Three Layered Neural Networks with Competition and Cooperation. International Workshop SPEECH AND COMPUTER, pp.111-114, October, Moscow, Russia, 1999.

Verbal Transformation effect by Two and Three layered Neural Network

Makoto Funamori, Tetsuro Kitazoe

E-mail: funamori@cs.miyazaki-u.ac.jp, kitazoe@cs.miyazaki-u.ac.jp

Department of Computer Science and Systems Engineering
Faculty of Engineering, Miyazaki University
1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192 Japan

ABSTRACT

We take a dynamical system to understand the auditory perceptual illusion known as the verbal transformation effect (VT). To take a step toward a neuronal dynamics, a system is considered to input data flow into a neural network and to output neural activities which show recognition result. The network is two and three layered and known to have cooperation and competition mechanism. Input data are made from those of two vowels pronounced separately. Though data are made artificially, they have a strong resemblance with the acoustic string to make VT effect since our data have a high similarity with two vowels at the same time. When the data was fed to 2LNN sequentially, the out-put signal showed clearly VT like effect; After a vowel was perceived for a while, other vowel became dominant and their role changed alternately in the following time.

1. INTRODUCTION

In 1958,Warren and Gregory had an interesting observation called verbal transformation (VT)[1,2]: When a person listens to a syllable, word or phrase repeatedly, the perception of the auditory stimulus is initially veridical. After a while, however, the subjects begin to perceive as changing among a variety of phonetic forms.

Recently many works focused on the temporal evolution of the phonetic changes and on the patterning of illusory forms[3]. Especially, Ditzinger et al presented a model world where similar phenomena occurs in the description of particle movement under a specific potential landscape. They assume the coordinates of the particle as representatives of order parameters in the statistical physics of neuronal many body systems[4].

In the present note we want to enter further inside the neural network system, aiming to process actual auditory input to a network and to get a verbal transformation output from the network. We present two or three layered neural networks (2LNN(3LNN)) [6,7,8,9] which have cooperation and competition mechanism and are able to process the phoneme recognition. Since a string of repeated auditory stimulus has ambiguous natures, it looks likes a repetition of a word or that of another. If one looks at such alternate forms, it is evident that they have such string that has similarity with some part of the sting. VT is understood that a neural network processes strong similarities with two or more syllables (or words) and recognize one form for a certain time span and then becomes to perceive alternately other phonetic forms in the subsequent time. At present, however, we do not know how to measure the similarities at the level of syllable or word. We intend to make a similarity table artificially made of repeatedly spoken vowels such that the table has strong similarity with two or more vowels at the same time. We are going to show how the neural network equations 2LNN(3LNN) exhibit VT phenomena when the similarity table made in this way is fed to the equations.

2. SPEECH RECOGNITION PROCEDURE

Before entering into discussion on VT, let us review our speech (phoneme) recognition system using 2(3) LNN neural net equations which is divided into four main processes;
(1) A number of training speech data are classified and parameterized into sequences of feature vectors for each phonemes. The feature vectors are used to form standard Gaussian PDFs which are supposed to be memorized in our brain for each phoneme.
(2) An input phoneme is referred to these memorized phoneme data and a similarity measure is obtained by comparing the input phoneme data with the memorized PDF of each phoneme.
(3) Suppose that there is a neuron activity ξ_u^a corresponding to the similarity measure λ_a^u to a certain phoneme /a/ at the frame number u.
(4) The neural net equations perform to make an activity ξ_u^a move toward a stable point after the equations receive the similarity measure as an input and a recognition results are achieved when it reaches to a stable state.
The memorized standard acoustic models for each phonemes are expressed in terms of Gaussian PDF for input o.

$$N(o;\mu_a,\Sigma_a) = \frac{1}{\sqrt{(2\pi)^n|\Sigma_a|}} e^{-\frac{1}{2}(o-\mu_a)^t \Sigma^{-1}(o-\mu_a)} \quad (1)$$

where μ_a is a mean value and Σ_a is covariance matrix, of feature vectors for training data of a phoneme /a/. The normalized similarity λ_a^u of input

data O_u at u-th frame to a certain phoneme /a/ is defined as

$$\lambda_u^a = \frac{\log N(o_u;\mu_a,\Sigma_a)-<\log N>}{<\log N>} \qquad (2)$$

where <logN> means an average over phonemes at the same frame u.

3. TWO (THREE) LAYERED NEURAL NET EQUATIONS WITH COMPETITION AND COOPERATION

The 2 Layered Neural Net(2LNN) equations are given as

$$\tau_1 \dot{\xi}_u^a(t) = -\xi_u^a(t) + f(\alpha_u^a) \qquad (3)$$

$$\tau_2 \dot{\alpha}_u^a = -\alpha_u^a + A\lambda_u^a - B\sum_{a'\neq a}g(\xi_u^{a'}(t))+D\sum_{u'=n-l}^{u+l}g(\xi_{u'}^a(t)) \qquad (4)$$

where $\xi_u^a(t)$ is a time-dependent neuron activity. f(x) is a well known sigmoid function and g(u) is a function given by

$$f(x) = (\tanh(w(x-h))+1)/2 \qquad (5)$$
$$g(u) = u^+ = (u+|u|)/2 \qquad (6)$$

where A,B,D,w,h are positive constants which are to be chosen appropriately .
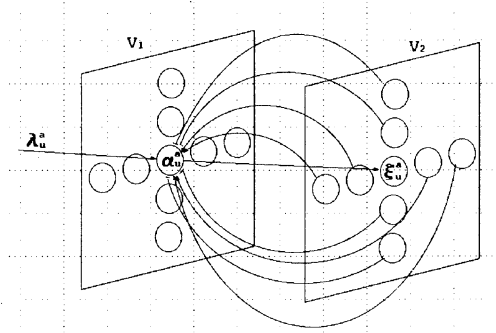
FIGURE 1 Neural network with two layered structure

of α and ξ

In equation (4), the second, third and forth terms in α_u^a are referred to the input, competitive and cooperative terms, respectively. The second term describes a similarity of input data at u-th frame to a certain phoneme /a/, the third term represents a competition with activities $\xi_u^{a'}(a'\neq a)$ of other phonemes and the forth term measures a cooperation of activities at the neighboring frames of the same

phoneme. The neural network for these equations has a two layered structure as shown in figure 1. At the equilibrium $\dot{\xi}_u^a=\dot{\alpha}_u^a=0$, the solution is determined independent of the initial values of ξ 's.

$$\xi_u^a(t) = f(\alpha_u^a) \qquad (7)$$

The solution shows that ξ_u^a is high for large values of α_u^a and low for small values of α_u^a .

The three layered neural net (3LNN) equations are given as

$$\tau_1 \dot{\xi}_u^a(t) = -\xi_u^a(t) + f(\beta_u^a) \qquad (8)$$

$$\tau_2 \dot{\alpha}_u^a = -\alpha_u^a + A\lambda_u^a - B\sum_{a'\neq a}g(\xi_u^{a'}(t))+D\sum_{u'=n-l}^{u+l}g(\xi_{u'}^a(t)) \qquad (9)$$

$$\tau_3 \dot{\beta}_u^a = -\beta_u^a + g(\alpha_u^a) + g(\xi_u^a) \qquad (10)$$

Since 3LNN gives similar results with 2LNN, we do not enter into details of 3LNN in the present paper.

4. ALGORITHM FOR CONTINUOUS SPEECH RECOGNITION

For continuous speech, it is considered that input data λ_a^u are fed to the neural network (3), (4) and the activities α, ξ in the network develop their value toward a stable point. We judge as that a phoneme /a/ is recognized at a flame u if ξ_u^a goes close to 1 while it is not recognized if ξ_u^a goes close to 0. To save calculation time in the present study, we assume $\tau_2 \ll \tau_1 = 1$ so that equation (3), (4) are simply written as

$$\dot{\xi}_u^a(t)= -\xi_u^a(t) + f(\alpha_u^a) \qquad (11)$$

$$\alpha_u^a = A\lambda_u^a - B\sum_{a'\neq a}g(\xi_u^{a'}(t))+D\sum_{u'=n-l}^{u+l}g(\xi_{u'}^a(t)) \qquad (12)$$

Notice that equation (11), (12) give the same solution as (3), (4) if the stable solution is unique. Differential equation (11), (12) are described as N steps of a loop in the numerical calculation divided by discrete time Δt and we obtain time development of ξ as

$$\xi(t),\xi(t+\Delta t),\cdots,\xi(t+N\Delta t) .$$

FIGURE 2 Recognition of /a/, /e/ changes alternately, showing VT effect. The parameter are set as A=1.5, B=1.7, D=0.4, h=1.0,w=3.0, l=2, Δt =0.05, N=200, L=10.

λ \	/a/	/e/	/i/	/o/	/u/
40	1.52	1.41	-0.12	-0.64	-0.46
39	1.45	1.43	0.35	-0.76	-0.57
38	1.50	1.19	0.42	-0.70	-0.45
37	1.41	0.56	-0.20	-0.12	-0.19
36	1.24	1.19	0.00	-0.68	-0.47
35	1.41	1.13	0.15	-0.67	-0.47
34	1.31	1.38	0.11	-0.69	-0.58
33	1.41	1.34	0.09	-0.65	-0.59
32	1.35	1.37	0.13	-0.69	-0.57
31	1.15	1.29	0.04	-0.63	-0.57
30	1.45	1.17	0.19	-0.63	-0.41
29	1.57	1.27	0.15	-0.66	-0.48
28	1.39	1.22	0.27	-0.62	-0.44
27	1.44	1.34	0.15	-0.64	-0.51
26	1.22	1.38	0.11	-0.66	-0.48
25	1.11	1.40	0.07	-0.64	-0.45
24	1.38	1.39	0.09	-0.68	-0.49
23	1.16	1.41	0.08	-0.66	-0.50
22	1.06	1.35	0.11	-0.63	-0.48
21	1.48	1.44	0.12	-0.67	-0.53
20	1.22	1.46	0.10	-0.66	-0.53
19	1.07	1.45	0.16	-0.67	-0.52
18	1.12	1.45	0.15	-0.68	-0.54
17	1.03	1.41	0.15	-0.67	-0.58
16	1.12	1.34	0.13	-0.67	-0.54
15	1.31	1.41	0.16	-0.70	-0.57
14	1.20	1.38	0.12	-0.68	-0.57
13	1.49	1.37	0.15	-0.69	-0.56
12	1.13	1.38	0.14	-0.68	-0.57
11	1.28	1.39	0.13	-0.67	-0.57
10	1.30	1.34	0.23	-0.66	-0.56
9	1.24	1.24	0.13	-0.60	-0.55
8	1.37	1.29	0.08	-0.63	-0.56
7	1.54	1.32	0.05	-0.65	-0.55
6	1.25	1.09	-0.23	-0.48	-0.50
5	1.28	1.03	-0.31	-0.42	-0.53
4	1.18	1.09	-0.26	-0.40	-0.34
3	0.74	1.47	-0.53	-0.52	-0.53
2	0.72	1.53	-0.71	-0.45	-0.62
1	0.51	1.85	-0.72	-0.54	-0.57
0	0.33	2.45	-0.67	-0.72	-0.72

TABLE 1 Synthesized similarity table which has high similarities with /a/ and /e/. The data are fed into the window from upper to lower, in which 2LNN processes the data inside the window.

To treat data flow λ_a^u sequentially, we consider that the data enter into a window with L frames and the neural network (11), (12) processes the L frames data for N steps. Then, the data are passed one frame forward through the window, where the initial values of ξ are set to zero before entering to the window. This is stated more in detail as follows. When input data λ_{u+L-1}^a, λ_{u+L-2}^a, ..., λ_u^a are entered inside the window, equations (11), (12) develop activities $\xi(t)$ until it arrives at $\xi_{u'}^a(t+N\Delta t)$ for $u'=u, u+1$, ..., $u+L-1$. Then, new data λ_{u+L}^a enter into the window from the above and the old data λ_a^u get out from the window as shown in table 1. At the same time whole $\xi_{u'}^a$ are shifted to $\xi_{u'-1}^a$, setting the initial values of $\xi_{u'-1}^a$ for the next N step loop calculations.

At this time we have new comers ξ_{u+L}^a entering into the window and the final ones ξ_u^a output from the window. We eventually judge if phoneme /a/ at u-th frame is recognized or not according to whether the final ξ is close to 1 or 0, respectively. A sequence of the same processes continues until whole input data go through the window completely and whole values of final ξ are obtained.

5. SIMULATION RESULTS FOR VT

We extracted a total 24 labeled phonemes from ATR Japanese speech database composed of 4000 words spoken by 10 male speakers and from ASJ speech database of 500 sentences by 6 male speakers to make Gaussian PDFs for each phoneme. The experimental conditions for acoustic feature parameters are

Sampling rate	16kHz,16bit
Preemphasis	0.97
Window function	16ms Hamming window
Frame period	5ms
Feature parameters	10-order MFCC +10-order Δ MFCC

To make the simulation of VT, we make two kinds of similarity table among five vowels /a/, /i/, /u/, /e/, /o/. We have two similarity tables when pronounced /a/ or /i/ for a few seconds. (We call /a/-table and /i/-table hereafter).When /a/-table was input into the neural

network (8), (9), we got a clear result such that ξ corresponding to /a/ became 1 for whole frames, while ξ for other vowels became zero. A similar result was obtained for /i/-table. For VT test, we synthesize two similarity tables into one. We plant the column /a/ of /a/-table into corresponding column of /i/ table. Thus, the synthesized table has large similarities for both /a/, /i/ as shown in table 1. We input this similarity data into 2LNN sequentially as described in section 4. The results are shown in figure 3 where we see a clear indication of VT phenomena, alternate recognition of /e/ and /a/.

6. CONCLUSION AND DISCUSSION

To understand VT phenomena from neurological aspects, we consider neural network equations 2LNN proposed recently to which input data are fed sequentially through a window. The input data made artificially have strong similarities with two kinds of vowels at the same time. The result shows clear VT phenomena, recognizing two vowels alternately.

To get a view of the mechanism of 2LNN, let us consider 2LNN (11), (12) applied to /a/-table. Since the stable solution does not depend on the detail of the initial value of ξ, we start the neural net equations by setting the same initial value for ξ's.
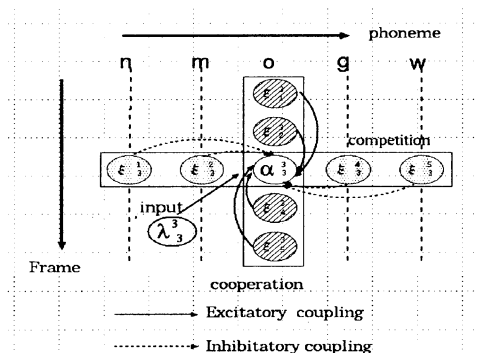
FIGURE 3   Dynamic process of neural activities for α neuron

At $t = 0$, α's take values corresponding to λ's. Then ξ's change their values through the sigmoid functions in equation (3) or (11). The α's are again influenced by excitatory coupling among neighboring

frame activities and inhibitory couplings from other phoneme activities as shown in figure 3. The cycle of the recurrent networks eventually gives ξ high or low values definitely in accordance with λ's values. In the case of /a/-table, phoneme /a/ becomes winner and others ore losers.

In the case of synthesized data (table 1), situations are something complicated. 2LNN tends to give high value for /a/ and /e/ at the same time. Since it is well known that the processes with cooperation and competition have a property called winner -take-all, ξ of either /a/ or /e/ compete each other to win definitely. The interesting point shown in the simulation is that a winner continue for a certain time span, then handing its to another vowel. Our next task is that we make a realistic similarity table to see how the neural network gives VT effect in the actual utterance.

Reference
[1] R.Warren, R.Gregory "An auditory analogue of the visual reversible figure" Am J Psychol Vol.71, pp.612-613, 1958
[2] R.Warren "Verbal transformation effect and auditory perceptual mechanisms" Psychol Bull Vol.70, pp.261-270,1968
[3] T.Ditzinger, B.Tuller, J.A.S.Kelso "Temporal patterning in an auditory illusion: the verbal transformation effect" Biol. Cybem Vol.77, pp.23-30, 1997
[4] T.Ditzinger, B.Tuller, H.Haken, J.A.S.Kelso "A synergetic modek for the verbal transformation effect" Biol. Cybem Vol.77, pp.31-40, 1997
[5] B.Tuller, J.A.S.Kelso "Characterizing the dynamics of auditory perception" CHAOS,Vol.5, pp.70-75, 1995
[6] T.Kitazoe, J.Tomiyama, Y.Yoshitomi, and T.shii "Sequential Stereoscopic Vision and Hysteresis" Proc. of Fifth Int. Conf. on Neurnal Information Processing, pp.391-396, 1998
[7] T.Kitazoe, Sung-Ill Kim,T.Ichiki "Speech Recognition by Stereo Vision Neural Net Pattern Recognition Equations" Joural of Artificial Life and Robotics Vol.4, 2000
[8] T.Kitazoe, Sung-Ill Kim,T.Ichiki "Acoustic Speech Recognition Model by Neural Net Equation with Competition and Cooperation" Proc. ICSLP Vol.7, pp.3281-3284,1998
[9] T.Kitazoe, Sung-Ill Kim,T.Ichiki, M.Fuanamori "Acoutic Speech Recognition by Two and Three Layered Neural with Competion and Coopertaion" Proc. SPECOM pp.111-114, 1999

Control System for Khepera Robot by Neural Network
with Competition and Cooperation

Kei Sugihara and Masayoshi Tabuse
Department of Computer Science and Systems Engineering
Faculty of Engineering, Miyazaki University
1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192 Japan
{ks, tabuse}@cs.miyazaki-u.ac.jp

Abstract

This paper describes a new approach to control systems for a mobile robot Khepera by using a neural network with competition and cooperation. Competition makes only one neuron active and cooperation keeps up the active state. In our researches, we find that the Khepera controlled by this neural network can draw a smoother trajectory than controlled by output values of Khepera's sensors themselves, especially in noisy environments.

1. Introduction

Recently, many people have been researching autonomous robots extensively. One purpose of this research is to build the robots that are able to behave in unknown or dynamically changing environments. The robots must make a plan and act by themselves, even if an unexpected occurrence should happen. One approach of this research is that the robots recognize surrounding environments and behave through the neural network[1].

In the present paper, we try to give a new approach to the neural network model by using competition and cooperation, which is used in a stereovision pattern recognition[2]. In the competitive and cooperative neural network, competition makes only one neuron active and cooperation maintains the active state. The robot recognizes the most important object in surrounding environments and keeps up this recognition under small fluctuations of sensor values, so that the robot behaves correctly in dynamically changing environments.

2. Khepera

In our experiments we use a miniature mobile robot Khepera[3]. The Khepera body is 32 mm height and 55 mm in diameter. It has two wheels, each of which is controlled by a DC motor, and can rotate in both directions.

The eight infra-red proximity sensors are installed around the Khepera body (six in front and two in rear, see Figure 1). These sensors allow two measures: the light reflected by obstacles and the ambient light. So these detect an obstacle and a light source. The output

values of each sensor are from 0 to 1023 in integer. For the measurement of the light reflection, 0 means that there are no obstacles near to the sensor, while 1023 means that an obstacle is very close to the sensor. The maximum detection range of the sensor is about 3 cm. On the other hand, for the measurement of an ambient light, the output values of the sensor degrease when the intensity of a light increases. The Khepera recognizes its own environments through these sensors, so that the Khepera can understand the local environments only.

The Khepera can communicates with a computer using a serial line. So we can obtain the sensor values from the Khepera and provide the wheel speed to the Khepera. In our experiments, the Khepera is controlled by a Sun SPARCstation through a serial line.
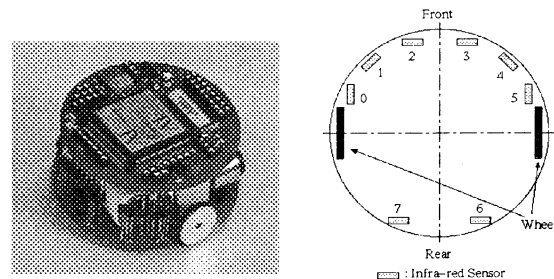
Figure 1: Mobile Robot Khepera

3. Three Layered Neural Network with Competition and Cooperation

We explain the 3 layered neural network[2], which controls the Khepera. The 3 layered neural network equations are given as

$$\frac{d}{dt}\alpha_u^a(t) = -\alpha_u^a(t) + A\lambda_u^a(t) - B\sum_{a'\neq a} g(\xi_u^{a'}(t)) + D\sum_{u'} g(\xi_{u'}^a(t))$$

(1)

$$\frac{d}{dt}\beta_u^a(t) = -\beta_u^a(t) + g(\alpha_u^a(t)) + g(\xi_u^a(t))$$ (2)

$$\frac{d}{dt}\xi_u^a(t) = -\xi_u^a(t) + f(\beta_u^a(t))$$ (3)

where ξ_a^u (t) is a neuron activity, $\alpha_u^a(t)$ is a first layer activity and $\beta_u^a(t)$ is a middle layer one. λ is an input value of this neural network. $f(x)$ is a well known as sigmoid function and $g(u)$ is a function given by

$$f(x) = (\tanh(w(x-h))+1)/2 \qquad (4)$$

$$g(u) = u^+ = (u+|u|)/2 \qquad (5)$$

where A, B, D, w, h are positive constants which are to be chosen appropriately. The neural network for these equations has a three layered structure as shown in Figure 2.
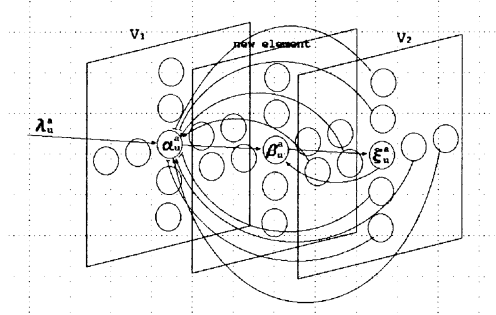
Figure 2: Three layered neural network

In equation (1), the second, third and forth terms are referred to the input, competitive and cooperative terms, respectively. The second term describes an input data, the third term represents a competition with other neuron activities $\xi_u^{a'}(a' \neq a)$ and the forth term measures a cooperation with other neuron activities ξ_u^a.

The qualitative feature of this neural network is as follows.

1) When all λ's have small values (about zero or less), all activities ξ are not active, i.e. their values are almost zero.

2) When one λ has a large value (between zero and one), its activity is active, i.e. its value is almost one.

3) When two or more λ's have large values, ξ of the largest λ is active and others are not active because of the competitive term.

4) When two or more λ's have very large values (about one or more), ξ of these λ's are all active.

5) Even if the value of λ for the active ξ becomes somewhat smaller, this ξ keeps up the active state because of the cooperative term.

6) According to the values of λ's, each ξ gets active or not.

Let us consider the control of Khepera by using this feature. In our experiments we suppose that the

neuron activities related to Khepera's sensors compete with each other and the time sequence of the neuron activity for each sensor cooperates with each other. Thus a and u indicate a sensor number (a=0,1,2,3,4,5,6,7) and the u-th past (u=0,1,...,10) at a discrete time step, respectively. The input value λ is defines as

$$\lambda_u^a = 2\,\frac{\text{sensor value of }\#a}{1024} - 1 \qquad (6)$$

for a light reflection and

$$\lambda_u^a = \frac{\text{med - sensor value of }\#a}{\text{max}-\text{min}} \qquad (7)$$

for a ambient light, where max, min, med are the maximum and minimum and median values of all sensors, respectively, and we calculate the neural network equations for a=0, 1, 2, 3, 4, 5, u=0 and keep up the u-th past activities (u=1,2,...,10).

4. Experiments

In order to estimate capability of the control of Khepera by the competitive and cooperative neural network, we consider the following 3 tasks:
· Obstacle avoidance
· Movement along a wall
· Approach to a light
and compare the results of these tasks with the case of the control of Khepera by the raw sensor data.

(1) Obstacle avoidance
We consider the obstacle avoidance task, which is that the Khepera moves around and avoids a wall (see Figure 5). The obstacle avoidance algorithm is

(i) Turn right if any ξ of the 3 front left sensors (sensor # = 0, 1, 2) is active.

(ii) Turn left if any ξ of the 3 front right sensors (sensor # = 3, 4, 5) is active.

(iii) Otherwise, go forward.

Figure 5(a) shows Khepera's trajectory in this experiment and the values of the Khepera's sensors and ξ are presented in Figure 3 and 4. The values of Khepera's sensors fluctuate due to the effect of the light from fluorescent lamps on the ceiling. On the other hand, the activities of ξ respond to a wall almost correctly. Figure 5(b) shows Khepera trajectory by using a raw data of sensors. Thus the Khepera controlled by the competitive and cooperative neural network can draw a smoother trajectory than controlled by the raw data of sensors.
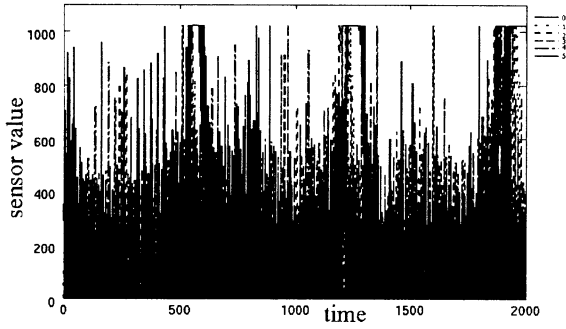
Figure 3: Sensor values of Khepera in experiment (1).
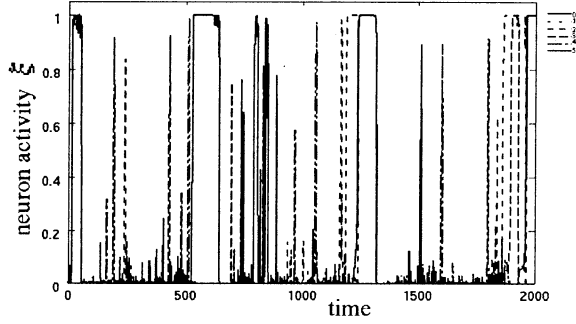
Figure 4: Neuron activity ξ in experiment (1).

(a)   (b)
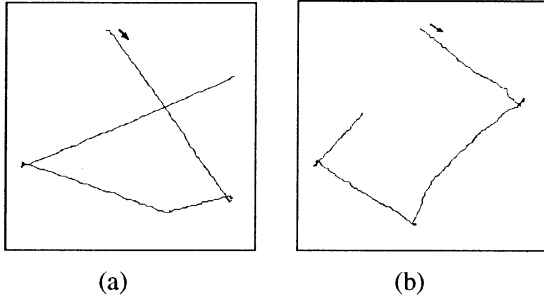
Figure 5: Khepera's trajectory in experiment (1) by using (a) the competitive and cooperative neural network and (b) raw sensor data.
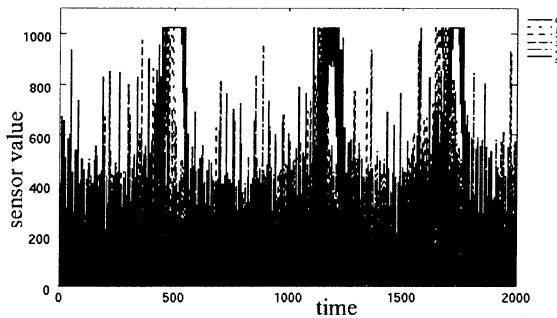
Figure 6: Sensor values of Khepera in experiment (1).

(2) Movement along a wall

We investigate the movement along a wall task, which is that the Khepera moves along a wall (see Figure 9). The algorithm of the movement along a wall is

(i) Turn left if any ξ of the 5 front sensors (sensor # = 0, 1, 2, 3, 4) is active, because there is a wall in front of Khepera.

(ii) Move forward if ξ of the #5 sensor is active, because there is a wall on the right side of Khepera.
(iii) Otherwise, move in the front right direction, because the Khepera loses sight of a wall.

Figure 9(a) shows Khepera's trajectory in this experiment and the values of the Khepera's sensors and ξ is presented in Figure 7, 8. From these Figures, we find that the Khepera moves along a wall successfully. Especially, at the corner, some ξ 's of the front sensors get active, even though ξ of the #5 sensor is active, so that the Khepera turns left according to the algorithm.

Next, we examine the Khepera's behavior in the case that a part of a wall is lacking as shown in Figure 9(b). In this case, the value of the #5 sensor decreases there. But the Khepera moves forward without turning right, because ξ keeps up the active state due to the cooperative term (see Figure 10, 11).
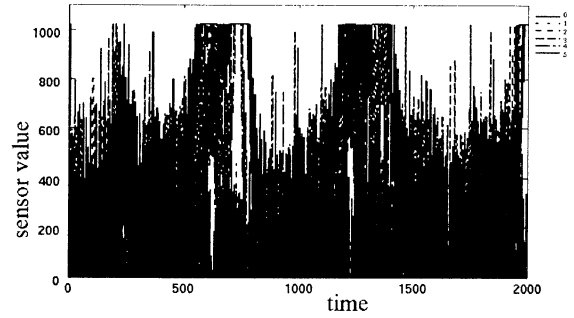
Figure 7: Sensor value of Khepera in experiment (2).
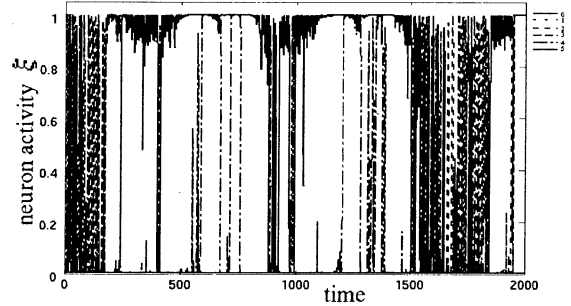
Figure 8: Neuron activity ξ in experiment (2).

(a)   (b)
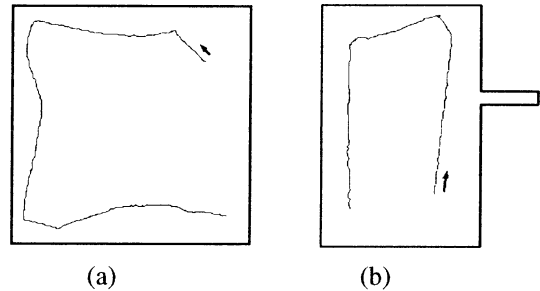
Figure 9: Khepera's trajectory in experiment (2).
　　　(a) Movement along a wall.
　　　(b) Movement along a wall, which is lacking partially.
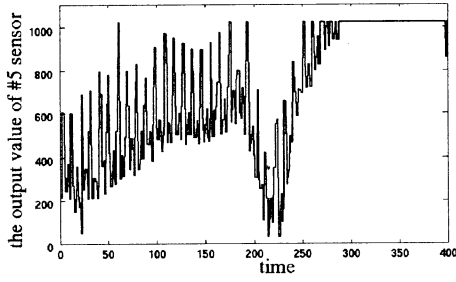
Figure 10: Output value of #5 sensor.
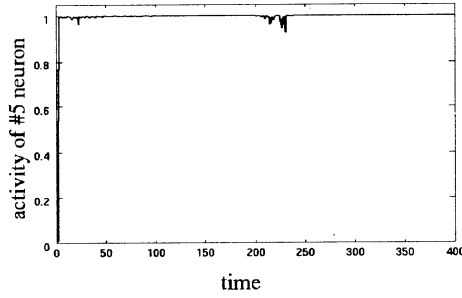
Figure 11: Neuron activity ξ related to #5 sensor.

(3) Approach to a light

We consider the approach to a light, which is that the Khepera avoids a obstacle and approaches to a light (see Figure 14). The approach to a light algorithm is

(i) Turn right if any ξ of the 3 front left sensors (sensor # = 0, 1, 2) is active.

(ii) Turn left if any ξ of the 3 front right sensors (sensor # = 3, 4, 5) is active.

(iii) Otherwise, go to the direction of active ξ.

Figure 14(a) shows Khepera's trajectory controlled by the neural network. The values of the Khepera's sensor of ambient lights and ξ are presented in Figure 12 and 13, respectively. On the other hand, Khepera's trajectory controlled by sensor values themselves is shown in Figure 14(b) and the sensor values are presented in Figure 15.
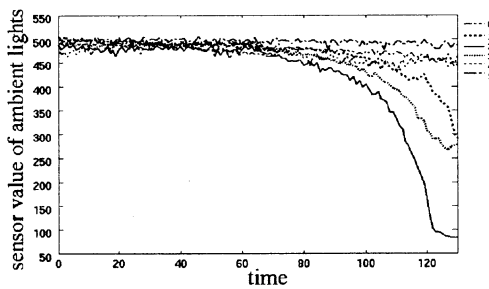
Figure 12: Sensor value of ambient lights in experiment (3).
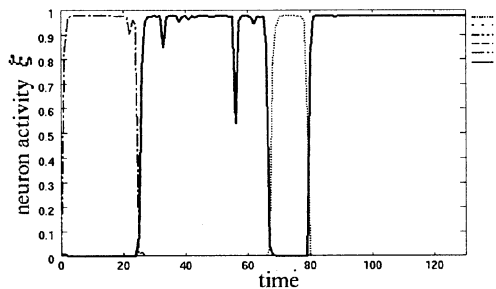
Figure 13: Neuron activity ξ in experiment (3).

(a)   (b)
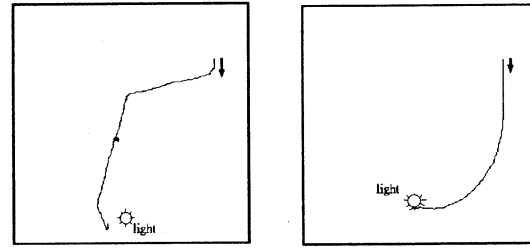
Figure 14: Khepera's trajectory controlled by (a) competitive and cooperative neural network and (b) raw sensor data.
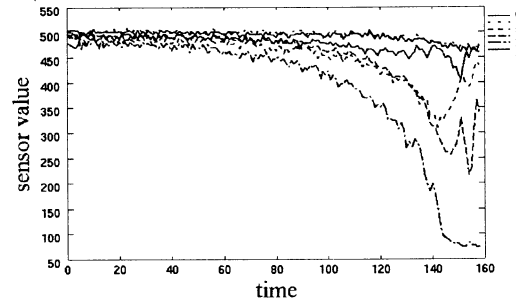
Figure 15: Sensor values of experiment (3).

5. Conclusion

We have presented the control system for mobile robots by competitive and cooperative 3 layered neural network and applied this system to the real robot Khepera. The competitive term makes only one neuron activity ξ get active for large input values and the cooperative term makes ξ keep up the active state for a small fluctuation of the input value. As a result, we could find that the Khepera controlled by the competitive and cooperative neural network can draw a smoother trajectory than by the value of sensors itself, especially in noisy environments.

Reference

[1] M.Murata, M.Nakamura, T.Ono and K.Onaga, "Behavior Planning of Autonomous Mobile Robots Based on Cooperative Neural Networks", Proc. of Third Int. Symp. on Artificial Life and Robotics (AROB III '98), pp.362-365, (1998).
M.M.Islam, Y.Wei, R.Odagiri, T.Asai and K.Murase, "Cooperation of Real Mobile Robots Using Communication", Proc. of The Forth Int. Symp. on Artificial Life and Robotics (AROB 4th '99), pp.309-312, (1999).

[2] T.Kitazoe, J.Tomiyama, Y.Yoshitomi and T.Shii, "Sequential Stereoscopic Vision and Hysteresis", Proc. of Fifth Int.Conf. on Neural Information Processing, pp. 391-396 (1998).

[3] "Khepera USER MANUAL Version 4.06", K-Team SA (1995).

Pattern Recognition of Emotional States using Voice, Face Image and Thermal Image of Face

Sung-Ill Kim, Yasunari Yoshitomi, Tetsuro Kitazoe

Department of Computer Science and Systems Engineering
Faculty of Engineering , Miyazaki University
1-1 , Gakuen Kibanadai Nishi , Miyazaki , 889-2192 Japan

E-mail : kim@eagle.cs.miyazaki-u.ac.jp

Abstract

A new integration method is presented to recognize the emotional expressions. We attempted to use both voices and facial expressions. For voices, we use such prosodic parameters as pitch signals, energy, and their derivatives, which are trained by Hidden Markov Model (HMM) for recognition. For facial expressions, we use feature parameters from thermal images in addition to visible images, which are trained by neural networks (NN) for recognition. The thermal images are observed by infrared ray which is not influenced by lighting conditions. The total recognition rates show better performance than each performance rate obtained from isolated experiment. The results are compared with the recognition by human questionairing.
Key words : Emotional Pattern Recognition, Emotional Speech, Prosody, Facial Expression, Thermal Image

1. Introduction

In the human-computer interaction, it will be a quite useful if a computer is able to recognize the emotional states which human beings express in the course of communication. Human-computer interfaces could be made to respond differently if the machine understands the emotional states of a user. Therefore, understanding those nonverbal communications has been one of the most important subjects for the ultimate goal to humanlike robot. Recently, many researches have been conducted on recognizing the individual nonverbal characteristics such as emotional factors contained in the speech, facial expressions, and body gestures etc.. However, there are still few reports considering and judging the human information totally.

In the present paper, we attempted to perform a new method modeling the emotional features and recognizing emotional states. We present an integration of human speech as well as visible and thermal facial expressions, aiming total understanding of the human mental states.

2. Emotional Feature Extraction

For recognizing emotional states in both voices and facial expressions, we need to extract emotional feature parameters from them. We first analyze voices which contain emotional information including four kinds of feature parameters. As well as emotional feature extraction from voice, we also extract useful feature parameters from facial expressions of both visible and thermal images.

2.1. Emotional Feature Extraction from Voice

The prosody[1,2] is known as an indicator of the acoustic characteristics of vocal emotions. In our experiments, we used four kinds of prosodic parameters, which consist of fundamental pitch signal (F0), energy, and each derivative elements. The pitch signals in the voiced regions were smoothed by a spline interpolation. In order to consider the effect of a speaking rate, furthermore, we also used a discrete duration information when training Hidden Markov Models (HMM).

We analyze the feature parameters from the speech waveform shown in figure 1, considering only the voiced regions as data points. All speech samples were labeled at the syllable level (/Ta/ and /Ro/) by manual segmentation in order to train each HMM using separated feature parameters. Figure 2 and 3 shows the pitch and energy signals extracted from each emotional speech of /taro/ spoken by a female student, respectively. In the figures, we can see that the feature signal of an emotion, anger, is the highest among other signals in each graph.
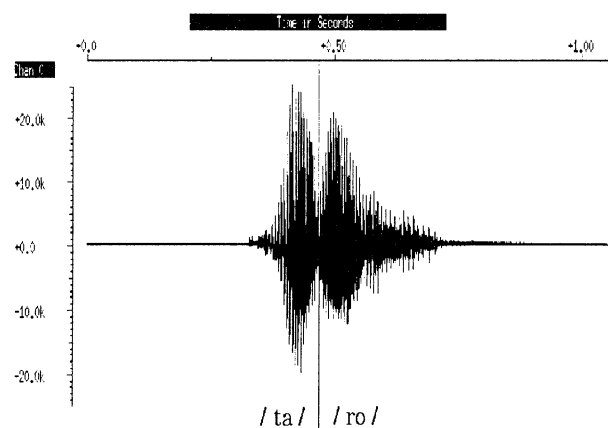
Figure 1. Speech waveform labeled by two parts /ta/ and /ro/.
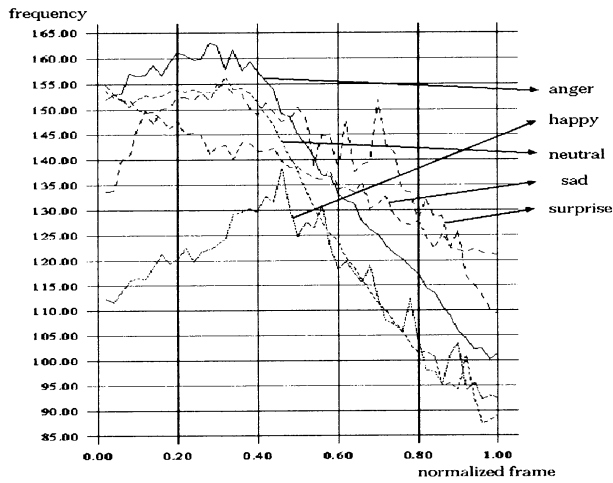
Figure 2. Pitch signals in each emotional state.
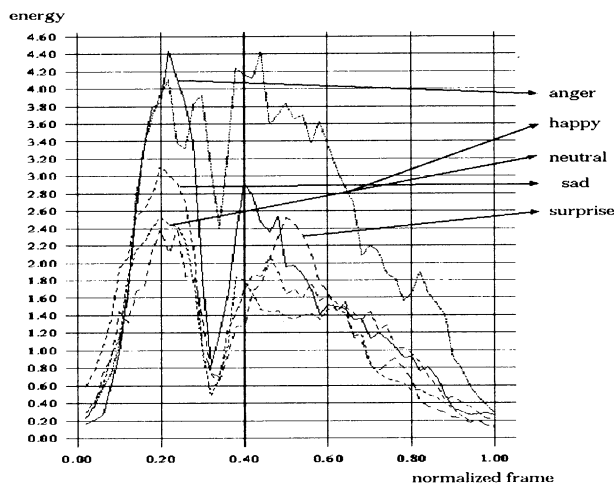
Figure 3. Energy signals in each emotional state.

2.2. Emotional Feature Extraction from Visible and Thermal Image of Face

Many studies have been performed to tackle the issues of understanding mental states of human through facial expressions, using ordinary visible camera. However, those trials still seem to be tough jobs since there is a only slight difference among various facial expressions in terms of characteristic features for the gray level distribution of input image using visible ray (VR). Thus, we have attempted to apply thermal distribution images to facial expression recognition[3,4] using an infrared ray (IR). Figure 4 illustrates the examples of female face images by VR and IR. VR image has the shortcoming that the accuracy of facial expression recognition is greatly influenced by a lighting condition including variation of shadow, reflection, and darkness. However, it is perfectly overcome by exploiting IR.

When a face image is given into recognizer, it is necessary to extract face-parts areas correctly, which will be important for better recognition performance.

Figure 5 and 6 show an extraction of face-parts areas which consist of three areas in the VR image and six

areas in the IR image, respectively.

Figure 4. Examples of face images of VR(a) and IR(b)
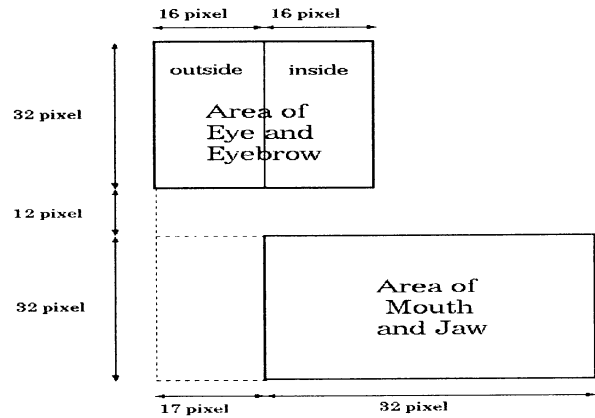
Figure 5. Extracting face-parts areas in the VR image.
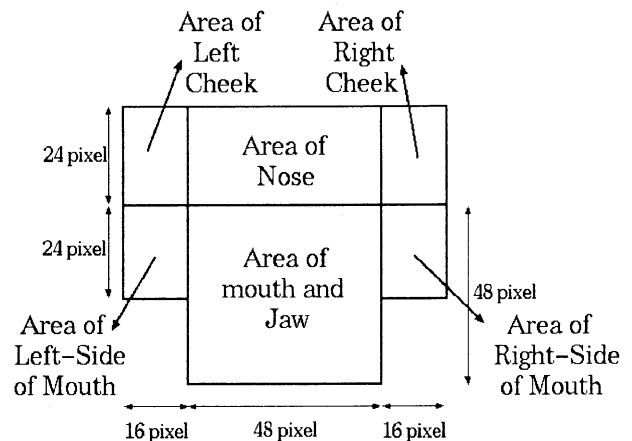
Figure 6. Extracting face-parts areas in the IR image.

In the next step, we generate differential images between the averaged neutral face image and the test face image in the extracted face-parts areas to perform a discrete cosine transformation (DCT). Figure 7 illustrates the procedure of extracting characteristic features from VR and IR images.

3. Recognition of emotion

In case of processing the emotional voice, the speech is sampled in the experimental conditions illustrated in table 1 for pre-processing of emotional voice recognition,

from which four dimensional emotional features are extracted.
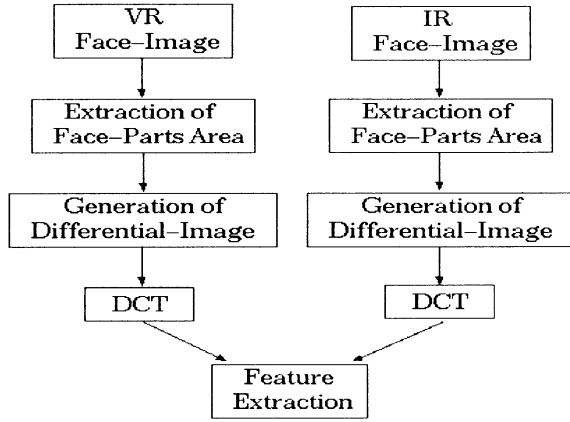
Figure 7. Procedure of emotional feature extraction from VR and IR image.

We then train the discrete duration continuous Hidden Makov Models (DDCHMM) by using these features with three states, using label information, and run recognition tests.

Sampling rate	16Khz , 16 Bit
Pre-emphasis	0.97
Window	16 msec. Hamming window
Frame period	5 ms
Feature parameters	pitch signal (F0), energy, delta pitch, delta energy, discrete duration information

Table 1. Analysis of speech signal

In case of processing the facial images, on the other hand, 57 and 78 bits of feature parameters are used as input data for neural network (NN) with three layers for IR and VR facial expressions, respectively. The unit number of hidden layer is decided experimentally for improving the recognition accuracy for facial expression. The unit number of output layer is the number of facial expressions which should be recognized.

Figure 8 shows the overall procedure of recognizing emotional states using integration method of three different recognition results. The integration of three separate recognition accuracies is performed by following equation.

$$S_i = \sum_{j=1}^{3} x_{i,j} \qquad (1)$$

where $x_{i,j}$ is an output value (1 or 0) for an emotional state i using a method j. Therefore, recognition results are chosen when the S_i is maximum.
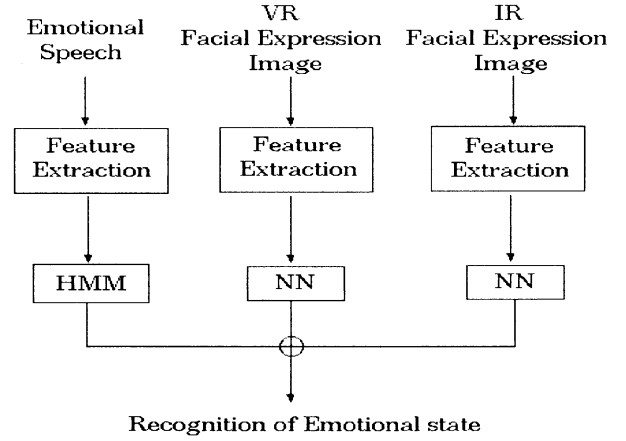
Figure 8. Procedure of recognizing the emotional state contained in voices and facial expressions.

4. Experimental Results

We first examine a human performance on three different types of questionairing. Next, we examine how effective our integration of three kinds of recognition methods is, when we compare the simulation performance results with human performance ones.

4.1. Database

The samples consisted of semantically neutral utterance, Japanese name 'Taro', spoken and acted by one actress. We capture voices and images simulating five emotional states such as neutral, angry, happiness, sadness, and surprise. We have simultaneously recorded voices and image sequences containing emotional states. We assembled 20 samples per each emotional expression as training data and 10 as test data.

4.2. Questionnairing Results

The emotional states of speech, image samples, and both of them are estimated subjectively by total 21 students who consist of 14 male and 7 female students. Table 2 shows the three kinds of human performance results we obtained. As shown in this table, the average recognition rates for five emotional states are 84.0%, 82.4%, and 92.5%, when using emotional voices, facial expressions, and both of them, respectively. From these human performances, we could see that our data were included relatively correct emotional states and that the questionairing result integrating both emotional voices and images gave better performance than those separately obtained from voices or images.

4.3. Simulation Results

We next performed recognition of mental states over the same test data used in the questionairing test, by integrating voices and facial expressions. Table 3 shows the recognition accuracies for each emotional state

in case of emotional voices, VR and IR facial expressions, and total recognition accuracies using the integration method of the three different recognition results, respectively.

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**86.2**	7.7			5.7
	Hap.	2.8	**75.7**	1.2	0.5	4.3
	Neu.		1.9	**97.1**		2.9
	Sad.	1.4	2.9	1.7	**99.5**	25.7
	Sur.	9.6	11.8			**61.4**

(a) Human performance on emotional voices

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**56.7**	4.0	2.7	2.7	5.3
	Hap.	3.3	**90.0**		3.3	2.7
	Neu.	15.3	2.0	**94.0**	2.0	11.3
	Sad.	21.3	2.0	2.7	**92.0**	1.3
	Sur.	3.3	2.0	0.7		**79.3**

(b) Human performance on facial expressions

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**95.3**	1.3			3.3
	Hap.		**92.0**		0.7	3.3
	Neu.			**100.0**	0.7	4.0
	Sad.		0.7		**98.7**	12.7
	Sur.	4.7	6.0			**76.7**

(c) Human performance on both emotional voices and images

Table 2. Human performance on each emotional state

As shown in the table, the average recognition rates for five emotional states are 60.0% 56.0%, and 36%, when using emotional voices, VR and IR facial expressions, respectively. In both cases of VR and IR facial expressions, the failure of recognition of emotion was mainly due to the difficulty to extract face-parts correctly

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**80.0**	20.0		20.0	10.0
	Hap.		**10.0**			
	Neu.		20.0	**100.0**	10.0	20.0
	Sad.				**50.0**	10.0
	Sur.	20.0	50.0		20.0	**60.0**

(a) Recognition accuracy for emotional voices

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**40.0**	40.0		20.0	30.0
	Hap.	50.0	**60.0**		10.0	
	Neu.			**70.0**		
	Sad.	10.0			**40.0**	
	Sur.			30.0	30.0	**70.0**

(b) Recognition accuracy for VR facial expressions

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**30**	20	10	10	20
	Hap.		**0**			10
	Neu.	20	10	**70**	10	40
	Sad.	40	60	20	**80**	30
	Sur.	10	10			**0**

(c) Recognition accuracy for IR facial expressions

		Input emotion				
		Ang.	Hap.	Neu.	Sad.	Sur.
Out put	Ang.	**50**	10		10	
	Hap.		**10**			
	Neu.			**80**		10
	Sad.	10			**50**	10
	Sur.	10	10		20	**40**
	No Ans.	30	70	20	20	40

(d) Total recognition accuracies using integration method

Table 3. Recognition accuracies for each emotional state

because the subject changed her face-orientation. Overall results are shown in table 3(d) and the total recognition rates amounts to 72% among five emotions (except for no answers).

5. Conclusion

This paper has described the new integration approach of recognizing human emotional states contained in voices and facial expressions. The emotional parameters were trained and recognized by HMM and NN for voices and images, respectively. The recognition results show that the integration method of recognizing emotional states in voices and images gives better performance than each isolated method in spite of the lower recognition rates compared to questionairing counterpart.

References

[1] Waibel, A (1986), Prosody and Speech Recognition. Doctoral Thesis, Carnegie Mellon Univ.
[2] C Tuerk (1990) A Text-to-Speech System based on {NET}talk. Master's Thesis, Cambridge University Engineering Dept.
[3] Yoshitomi Y, Miyawaki N, Tomita S, and Kimura S (1997), Facial Expression Recognition using Thermal Image Recognition and Neural Network. Proc. of 6th IEEE Int. Work. on Robot and Human Communication, pp.380-385
[4] Sugimoto Y, Yoshitomi Y, and Tomita S (in press), A method for Detecting Transitions of Emotional States using a Thermal Facial Image based on a Synthesis of Facial Expressions. Journal of Robotics and Autonomous Systems

Quantitative Analyses and Fractal Structures of Fish School Movements

Tatsuro Shinchi*,Haruhiko Nishimura** ,Tetsuro Kitazoe*** ,Masayoshi Tabuse***

*Dept. of Education and Culture,Miyazaki Univ.,1-1 Gakuen-Kibanadai-Nishi,Miyazaki City,889-2192 Japan
**Studies of Information Science,Hyogo Univ. of Education,942-1 Yashiro-cho,Hyogo,673-1494 Japan
***Dept. of Engineering,Miyazaki Univ.,1-1 Gakuen-Kibanadai-Nishi,Miyazaki City,889-2192 Japan

Abstract

The schooling of fish is the typical aggregate movements like as flocks and herds, and offers a considerable survival advantage. The gliding fish schools in union behave like a single organism, despite in the absence of a consistent leader for the entire system. In this paper, we analyzed the properties of autonomous fish school movements using the computer simulation model based on the observation data.

By comparison swimming trails depicted with multiple time-spatial,we found movements of fish schools possess structural self-similarity.And we grasped these properties of fish school movements as the fractal dimensions.

1 Introduction

Fish schools are the typical aggregate movements,reduce the probability of detection by a predator and reduce the risk of being eaten once the school has been detected.The fish school in union behave as one body like they have the intelligence for all fishes,despite in the absence of a consistent leader[1]. Such an autonomous behavior or a self-organization phenomenon has been studied in the field of ecology or zoology.However,it is too hard to grasp precisely all individual's behavior under the natural conditions,because animals move vary widely.In recent yeares,computer simulations are developed to understand the properties of complex behavior.And we can examine the animal movements as the complex adaptive systems,emerging by cooperative interactions among individuals in a group.

In this study,we found the two-dimensional coordinates of all fishes simulated in the computer systems.We grasp consequently the individual fish behavior changing momentarily as time-series data.This paper is intended as an introduction of fractal analyses to evaluate quantitaively the autonomous fish school movements.

2 Behavior Model of a Fish

The schooling of fish has been observed and investigated in various approaches to understand the rules of forming groups and maintaining groups.These researches revealed that most important senses for fish schools are the eyes and the lateral lines which detects water currents, vibrations, and pressure changes.It is known that the visual angle of fish eyes is often larger than 300° and perception angle of the lateral lines are similarly high.
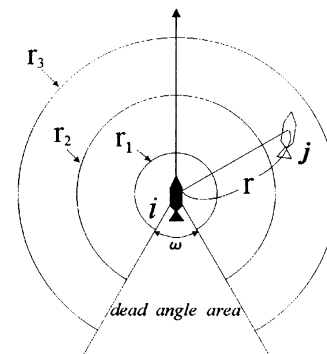
Fig. 1: Rages of the basic behavior patterns

Fish school movements emerge despite each individual lacking knowledge of movements for the entire school.It follows that the individual interactions with the perceived neighbors resolve a fish movement[2].On this study,we simulate schooling phenomenon based on two factors,mutural attraction and parallel orientation.Aoki suggested the behavior models,in which

four styles of reactions according to different possible positions of neighbor's:repulsion,parallel orientation,attraction and searching[2].

Fig.1 shows the geometrical drawing to illustrate parameters specifying interactions.A direction of the black fish(i) located in the center will be determined by the distance r to the perceived white fish(j) as follows,

i)repulsion($r < r_1$):
[If the neighbor fish(j) is too close,the fish tries to avoid a collision.],

ii)parallel orientation($r_1 \leqq r < r_2$):
[If the neighbor fish(j) is in a certain parallel orientation area, the fish swims in the same direction as its neighbor.],

iii)attraction($r_2 \leqq r < r_3$):
[If the neighbor fish(j) is too far away,the fish swims in the direction of its neighbor.],

iv)searching($r > r_3$ or dead angle area):
[If the fish cannot perceive its neighbor, the fish begins to search for a neighbor fish with turning around by chance.].

In each action i) $\sim$ iv) at time t,the fish(j) turns by the angle $\beta_{ij}(t)$ as follows,

i)repulsion($r < r_1$):
$$\beta_{ij}(t) = min\{\angle(\overrightarrow{v_i(t)}, \overrightarrow{v_j(t)}) \pm 90°\},$$
ii)parallel orientation($r_1 \leqq r < r_2$):
$$\beta_{ij}(t) = \angle(\overrightarrow{v_i(t)}, \overrightarrow{v_j(t)}) ,$$
iii)attraction($r_2 \leqq r < r_3$):
$$\beta_{ij}(t) = \angle(\overrightarrow{v_i(t)}, p_j(t) - p_i(t)),$$
iv)searching($r > r_3$ or dead angle area):
$$\beta_{ij}(t) = 0 .$$

abbreviations

$\angle(\vec{a}, \vec{b})$:an angle between two vector $\vec{a}$ and $\vec{b}$

$min\{a \pm b\}$:a smaller angle,$a + b$ or $a - b$

p_j:a coordinate of the fish(j)

It results that the new direction $\alpha(t + \Delta t)$ of fish(i) is

$$\alpha(t + \Delta t) = \alpha(t) + \beta_{ij}(t) + \sqrt{2}\beta_0. \qquad (1)$$

The term of $\sqrt{2}\beta_0$ is a fluctuation in determining a new direction,β_0 follows a Gamma distribution will be found in a range $-15.0 \leqq \beta_0 \leqq 15.0$.The neighbor fish($j$) among N fishes which will be selected with the inverse proportion to the distance r,and the nearer fish will be perceived as the fish(j).

3 Simulation of Fish School Movements

With Aoki's model,we can reproduce the various types of behavior by the changing the range r_1,r_2,and r_3(Fig.1).Huth and Wissel reported that reproducing the highly polarized school needs $0.3BL \leqq r_1 \leqq 0.5BL$ and $1.0BL \leqq r_2 - r_1 \leqq 1.5BL$($BL$:Body Length)[4]. The polarization,the one of striking qualities of fish schools, means the parallel arrangement of the members.Aoki called *Standard Run* the polarized fish school with $r_1 = 0.5BL$, $r_2 = 2.0BL$,and $r_3 = 5.0BL$, which are in the ranges reported by Huth.
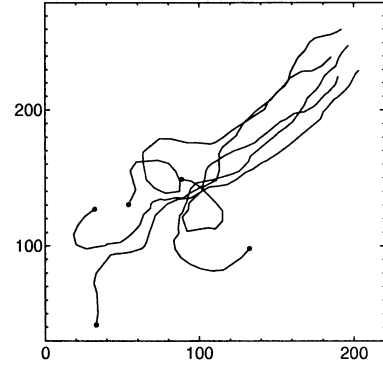
Fig. 2: The self-organization of a fish school(N=5), $t = 1 \sim 50$

In this paper,we fixed the model with $r_1 = 0.5BL$, $r_2 = 2.0BL$, $r_3 = 5.0BL$,and $\omega = 30°$(Fig.1). One simulation step in this model is set to 0.5[s] ,correspond to the earlier simulation studies for fish school movements[3][4].Fig.2 shows that the movements of all five fishes in beginning 50 simulation steps.Each fish self-organizes gradually the parallel arrangement and swims with forming a school to the upper right-hand corner of Fig.2,even if their directions and positions are initialized randomly.

Fig.3 shows the trail of one fish in the school consisted with 5 fishes under the setup of *Standard Run*,which is obtained by the run for 10000 simulation steps duration corresponding to 5000 [s] duration. It assumed that even other individuals almost draw the same locus in the polarized schools, except for the period when it needs to the self-organization immediately after a simulation start.Therefore,it can be said that the evaluation to the behavior of one tail results the analysis of school movements.In this paper,we examine the character of the complicated fish school movements seeing in Fig.3.
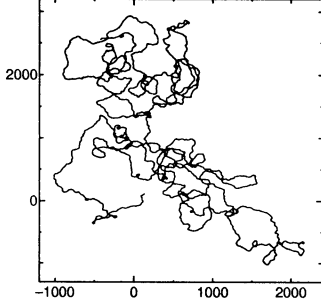
Fig. 3: The trail of fish school(N=5),$t = 1 \sim 10000$

4 Fractal Structures

The evaluation to the complicated structures and the behavior observed in natural phenomena have been progressed with the concept of fractal.In a fractal analysis,an object is calculated with plural different lengths.We gain the characteristics of the irregular fish school movements in Fig.3 with plural time scales,which means the plural levels of coarsening.

Fig.3 shows the time-series trail pattern that all position coordinates are connected with straight lines ($\Delta t = 1$).Fig.4(a) is the trail pattern coarsened with every 10 step($\Delta t = 10$),which is the part ($t = 5000 \sim 5500$) of Fig.3 and consists of 50 straight lines.Furthermore,Fig.4(b) is the coarsened trail pattern,consists of 50 straight lines every 20 step($\Delta t = 20$),becomes 50% definition of Fig.4(a).

Similarly, Fig.5(a) is the coarsened trail pattern which consists of 50 straight lines every 100 step($\Delta t = 100$),the part of the second half 5000 step of Fig.3.And Fig.5(b) is the coarsened trail pattern ,consists of 50 straight lines every 200 step($\Delta t = 200$).

Both of Fig.4 and Fig.5 shows that Fig.(a) and Fig.(b) are resembled well mutually,in spite of the difference of coarsening level.It indicates that some of fish school movements observed from different coarsening level are self-similar on multiple scales,in that a small part will look similar to whole.Namely,fish school movements have the property of Fractal.The self-similarity is the one of interesting qualities of Fractal.

On the other hand,In the comparison of the coarsened trail patterns between Fig.4 and Fig.5,Fig.5 is clearly more complicated structure.It results that the trail pattern has the property corresponds to the level of coarsening. As above,even if it is the one fish school movement,they show different aspects with the observation level.
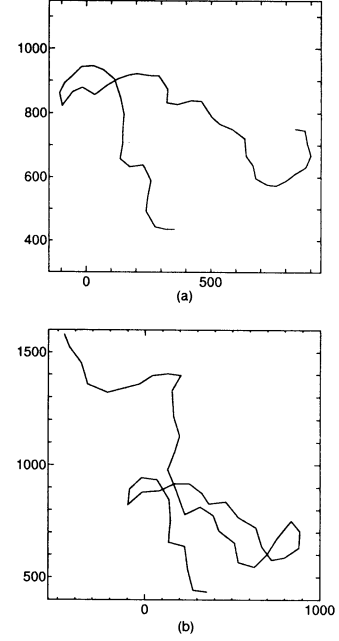
Fig. 4: The coarsened trail pattern I (with 50 lines) $(a) : t = 5000 \sim 5500(\Delta t = 10)$,$(b) : t = 5000 \sim 6000(\Delta t = 20)$
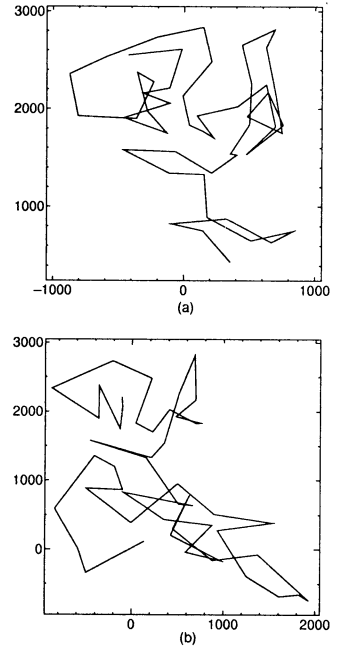
Fig. 5: The coarsened trail pattern II (with 50 lines) $(a) : t = 5000 \sim 10000(\Delta t = 100)$, $(b) : t = 1 \sim 10000(\Delta t = 200)$

5 Fractal Analyses

To evaluate quantitatively the difference of complexity and similarity caught visually in Fig.4 and Fig.5,we'll find the fractal dimension of the fish school movements with the time-series coordinates of a fish.The coarsened length $< L(k) >$ of the fish trail pattern is computed by a line segment with each regular time interval k[5][6].

$$< L(k) > \quad = \quad \frac{1}{k}[\sum_{t=1}^{T-k} \sqrt{(x_{t+k} - x_t)^2 + (y_{t+k} - y_t)^2}$$
$$\cdot \frac{T}{(T - k)k}] \qquad (2)$$

The whole is multiplied by $1/k$,because $< L(k) >$ is found from k sets of time-series with each the different starting point in computing. T is the all time of the fish school simulation.And the term of $T/((T - k)k)$ represents the normalization factor for the length of subset time-series. In the case that the following relation

$$< L(k) > \propto k^{-D} \qquad (3)$$

materializes, D can be called the fractal dimension.

The coarsened length $< L(k) >$ becomes shorter accompanied by the increase of k,and $< L(k) >$ decreases largely from $k = 60$ neighborhood.The absolute value fit to the gradual inclination plot group $(1 \leq k \leq 60)$,and to the urgent inclination plot group $(60 \leq k \leq 400)$,becomes $D_1 = 1.08$,and $D_2 = 1.42$ individually(Fig.6). In Fig.6,( ) inside is the coefficient of correlation in fitting.D_1 and also D_2 are fractal dimensions obtained by the evaluation the property of the fish school movement pattern observed in Fig.4 and Fig.5 from the different coarsening level.

6 Summary

We evaluated the fish school movements on the computer,in the period exceeds one hour in real time,too long time to observe in the river or the sea.And the result of quantitative fractal analyses for the long period behavior,it became clear that the fish school movements will have the fractal property,responding to the levels of coarsening.Namely, this study shows that the polarized fish schools are the fractal systems,which is said to excels in error tolerance[7].It's speculated that the usual school movement which fractal property have can be changed smoothly to the instantaneous target movement for
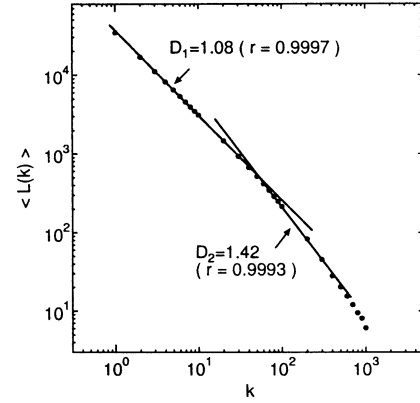
Fig. 6: The fractal analyses

a special purpose,the bait search action, the escape action, and the reproduction action etc.

We might go on to the fractal analyses for instantaneous target movements and the movements of other animals.

References

[1] Braian L.Partridge:The Structure and Function of Fish Schools, Scientific American,vol246, pp.90-99(1982)

[2] Ichiro Aoki:An analysis of the schooling behavior of fish:Internal organization and communication process,Bull.Ocean Res.Inst.Univ. Tokyo,12,pp.1-65(1980)

[3] Ichiro Aoki:A Simulation Study on the Schooling Mechanism in Fish, Bulletin of the Japanese Society of Scientific Fisheries,48(8), pp.1081-1088(1982)

[4] A.Huth and C.Wissel:The Simulation of the fish schools in comparison with experimental data,Ecological Modeling 75/76,pp.135-145(1994)

[5] Tomoyuki Higuchi:Approach to an Irregular Time Series on the Basis of the Fractal Theory,Pysica,vol.D31,pp.277-283(1988)

[6] H.Nishimura,T.Shinchi : Fractal Analysis of One-dimensional Cellular Automata as Time-series Vectors,Information Processing Society of Japan Transactions,vol.36,no.4,pp.787-796(1995)

[7] West BJ : Fractal Physiology and Chaos in Medicine, World Scientific,Singapore (1990)

Contextual Modulation with Reinforcement Learning in a Temporal Coding Neural Network

Masataka Watanabe †*and Kazuyuki Aihara* †††††

Email:watanabe@sk.q.t.u-tokyo.ac.jp

†Dept. of Quantum Engineering and Systems Science, Tokyo University
††Dept. of Mathematical Engineering and Information Physics, Tokyo University
†††CREST, Japan Science and Technology Corporation (JST)

ABSTRACT

We propose a neural network model which is capable of switching its functions with the change of context. In this model, functional connectivity, the dynamic neuronal interaction in temporal pulse coding, is the key to such characteristics. Furthermore, we introduce two learning rules to our model, temporal hebbian rule and reinforcement learning. The former works to self organize the dynamics generated by the context and pattern input, while the latter couples the self organized dynamics and the preferred response.

KEYWORDS: temporal pulse coding, functional connectivity, contextual modulation

1. Introduction

Over the past few decades, researches of modern scientists have thrown new light on the nature's most highly complicated system, the brain. It is true that much fascinating truth of the brain has been revealed due to the current progress in experimental technique and computer simulation. Although, a very fundamental problem remains an open question:*what is the carrier of information in the brain?*

It is quite obvious that the brain makes use of spikes transmitted among neurons through synaptic connections for information processing. The question is what constitutes information in these spike trains. Is it the mean firing rate or the timing of spikes? This is a key question that must be solved to understand the information processing and the knowledge representation of the brain.

The traditional view to this question is that mean spike rate in a psychological time scale of hundreds of milliseconds is the carrier of information. Many researchers came to believe this hypothesis due to an experimental evidence that the torque generated in muscles was proportional to the spiking rate in motor systems [1]. Most physiological experiments and neural network models are based on the rate coding hypothesis.

However, the possibility of temporal pulse coding, which assumes the spike timing as the carrier of information, has been recently brought to light by Malsburg [2] . His theoretical viewpoint was that timing of spikes plays an essential role in encoding, representing, and processing information and knowledge in the brain. This pioneering work was followed by physiologist seeking for proof of temporal pulse coding in the actual brain. Since the most significant difference of rate coding and temporal pulse coding is whether the neuron functions as integrators or coincidence detectors, considerable numbers of attempts were made to clarify this question. Softky and Koch [3] reported that the inter spike intervals(ISI) of cortical pyramidal cells are highly variable, which is consid-

ered as a circumstantial evidence of these neurons to function as coincidence detectors. Although a counter-argument was made by Shadlen and Newsome [4] that the irregularity of the ISI may be explained by balance of excitatory and inhibitory incident spikes, Softky [5] came up with an attractive explanation: the effective coincidence detector hypothesis. This hypothesis claims that the presence of balanced excitatory and inhibitory inputs results to the functioning of an integrator neuron as a coincidence detector.

Apart from the question of the function of cortical neurons, Vaadia, Aertsen, Abeles and their colleagues have observed dynamical modulation of temporal correlations in a short time-scale of the order of 100 ms in a task-related manner [6]. This result is considered as a more direct proof of temporal pulse coding in cortical information processing. There are still other experimental work which suggest the plausibility of temporal pulse coding in the brain [7] [8].

While the heated debate of whether the brain is using temporal coding or rate coding or even both is pursued, we look at this problem from another view point. We consider what new tools can temporal coding provide to neural network models. Our opinion is that one of the keys of temporal coding is the dynamic neuronal interaction, namely, "functional connectivity".

In this paper, we first give a brief description of functional connectivity in contrast with classical synaptic connectivity in rate coding and next move on to the central to this issue, *what it can do*. Here, we come up with a new concept, that is to say, neural networks which are capable of switching its functions by "software" [11],[12]. The essential to this model is functional connectivity and the two learning rules, "reinforcement learning" and "temporal hebbian learning". Finally we conclude with discussions.

2. Functional Connectivity

To begin with, the most significant difference between rate coding and temporal pulse coding is based upon the behavior of neurons. The decay constants characterize the behavior of the neurons. A neuron in rate coding has a large decay constant, so that the neuron acts as an integrator of incident pulses. On the other hand, the decay constant of a neuron with temporal pulse coding is very small, and the neuron functions as a coincidence detector of input pulses. In this case, the timings of input pulses become a significant factor for information processing.

Now that we are ready, we consider the difference of interaction between neurons in rate coding and temporal pulse coding, namely, the classical synaptic connectivity and functional connectivity. In the case of rate coding, pulses from

a pre-neuron give almost equal influence to the post-neuron irrelevant to input timing; thus, the neuronal interaction in rate coding is static. In contrast, interaction in temporal pulse coding is dynamic because the influence of a single pulse to a neuron depends on input timing with other incident pulses due to small decay time constants. We can say that the strength of connection is determined not only by the synaptic efficacy but also the ongoing network dynamics. Such dynamic connection in temporal pulse coding is called functional connectivity [2] [13].

3. Models of Neurons and Neural Networks
3.1. Neuron Model

The neuron model used in this research is a simple continuous-time model with two parameters; the decay time constant τ of the neuronal internal state and the firing threshold θ (Judd and Aihara 1993, Ichinose and Aihara 1995). A neuron receives incident pulses asynchronously with continuous interspike intervals and whenever the internal state exceeds the threshold, the neuron fires and sends out a pulse to all neurons to which it is connected. The change of the internal state is described as follows,

$$a(t_{n+1}) = s_{n+1} + a(t_n)exp(-T_{n+1}/\tau) \qquad (1)$$

where t_n is the arrival time of the nth incident pulse, $a(t_n)$ the value of the internal state of the neuron at time t_n, s_{n+1} the strength of the $(n + 1)$th pulse and T_{n+1} the interspike interval between the nth incident pulse and the $(n + 1)$th incident pulse. In this paper, we fix the values of most s_n at 1 for the sake of simplicity and concentrate upon spatio-temporal structures of excitation dynamics. The neuron fires with following probability,

$$P^f = sigmoid(a(t_{n+1}) - \theta) \qquad (2)$$

and if the neuron fires, it sends out a pulse to all neurons which it is connected and the internal state $a(t_{n+1})$ is reset to 0.

3.2. Network Model
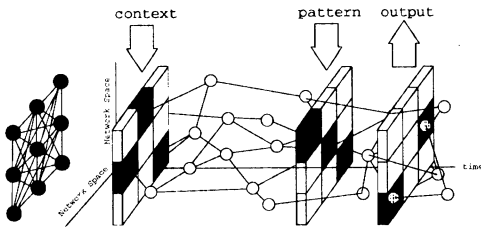
Figure 1: Network model

In this subsection, the architecture of our model is illustrated(see figure; the connections among neurons and formats of program input, a spatial pattern input and a spatio-temporal output pattern.

The neural network is composed of 64 neurons mutually connected with finite delays; every neuron is fundamentally connected to all other neurons and it requires finite time for pulses to propagate from a neuron to a neuron. In this model, the delays are set randomly according to a normal distribution with mean delay $\bar{d}$ and variance $< d >$. Once the delays are set, they are fixed throughout the simulation.

Another important factor concerned with connections among neurons is that values of all connection weights are set to 1 except for the program input which is described later. It is meaningless to have such uniform connection strength in conventional neural networks, but since our network uses functional connectivity instead of conventional synaptic connectivity, it is possible to handle information even with these uniform connections.

Turning now to the context input, it plays an important role in our network model. The context determines the dynamics of the network and hence the functional connectivity. In this study, the context input correspond to a spatio-temporal spike pattern given to the network. The connection weights to these neurons from the program input is strong enough so that it can fire these neurons by itself.

Next, the pattern input to the network is in the form of spatial suppression of neurons. We choose the neurons to suppress according to the given spatial input pattern as shown in figure (1). This suppression of neurons is equivalent to receiving high frequency pulses through inhibitory connections.

The output is given in the form of a rate coding firing pattern of neurons in the network. The details are given in the next section.

Finally, we will take into account a stabilizing mechanism in our model. Pulse-coded neural networks show very unstable behavior that without any stabilizing device, all neurons will easily become quiescent or firing with the highest rate(Abeles 1991). In this study, we use a global stabilizer that will keep the number of firing neurons in the network to a moderate rate. What we mean by "global" is that we give negative feedback of the total activity in the network to each neuron. The stabilizer calculates its internal value S as follows,

$$S(t_{n+1}) = 1 + S(t_n)exp(-T_{n+1}/\lambda) \qquad (3)$$

where t_n is the time of the nth firing in the network, T_{n+1} the interval time between the nth firing and the $(n + 1)$th firing in the network and λ the decay time constant. We negatively feedback the proportional value $\alpha S(t)$ to every neuron in the network, where α is the proportionality constant and $S(t)$ for time t between successive firings in the network is given as follows,

$$S(t) = S(t_{n^*})exp(-(t - t_{n^*})/\lambda) \qquad (4)$$

where t_{n^*} is the time of the nth firing in the network. The actual threshold of each neuron is defined as follows,

$$\theta(t) = \theta_0 + \alpha S(t) \qquad (5)$$

where θ_0 is the basic threshold.

3.3. Learning rules

We introduce two learning rules to our model. First is the temporal hebbian learning rule. This learning rule was
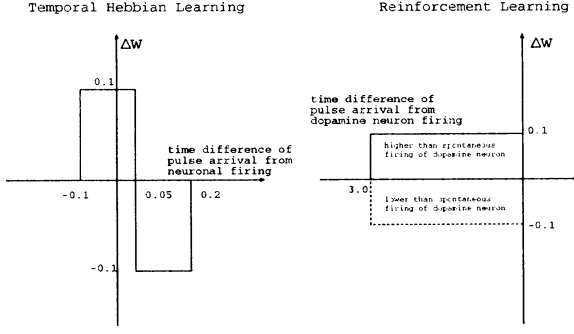
Figure 2: Two learning rules: temporal hebbian learning rule and reinforcement learning

Figure 3: Context and pattern input: from the top, AX, AY, BX, BY.

first proposed by Gerstner et al.[9] and later confirmed by Markram et al.[10]. The key to this rule is that whether the synaptic efficacy increases or decreases depends on the precise timing of the spike arrival and neuronal firing. If the spike arrives faster than the firing, the efficacy increases and if later decreases. Such learning rule plays the role to self organize the dynamics produced by the interaction of context and pattern input. Since our network model is composed of stochastic neurons, before the dynamics is self organized, even with the same context and pattern input, there is always different dynamics. After given the same set of context and pattern input for several times, the dynamics will be self organized.

The second learning rule, the reinforcement learning, works to connect the self organized dynamics with the desired output. This rule is very different from others that the sign of efficacy change is determined by the future firing of the dopamine neuron. If the synapse receives a spike input and followed by a dopamine release, the synaptic efficacy is increased. On the other hand, if there is no dopamine release the efficacy is decreased. There are number of hypotheses on what condition the dopamine neuron fires, but it is sure that in the early phase of learning, it reacts to the presence of reward. In our model we assumed that the dopamine neuron fired when the output was correct. The change of synaptic efficacy goes back in time for 3.0 unit time.

4. Simulation Results

4.1. Context, pattern and output: before learning

Figure (3) shows the combination of context input and pattern input with the network dynamics before learning. We used two kinds of context intput A, B and also two kinds of pattern input X,Y. We can see that context input is given as a spatio-temporal spike input and the pattern input is given as spatial suppression of neurons. The output of the neuron is determined by counting the numbers of neuronal firing in a certain time window. If the number of neurons firing with the first half of the index is larger, the output is assumed as No-GO, and else GO. The desired response is as the following. AX → GO, AY → No-GO, BX → No-GO and BY → GO.
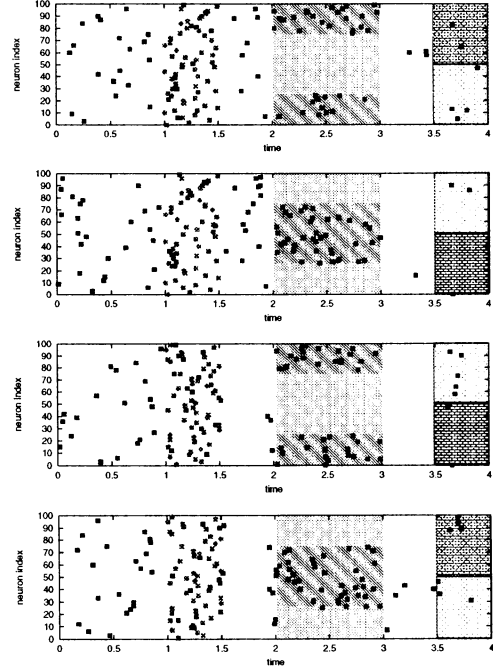
4.2. Learning process

Figure (4) gives the process of learning with the rate of correct response. The learning process was repeated as follows, AX 20, BX 20, AY 20, AX 20. The rate of correct response increased as the learning process went on.
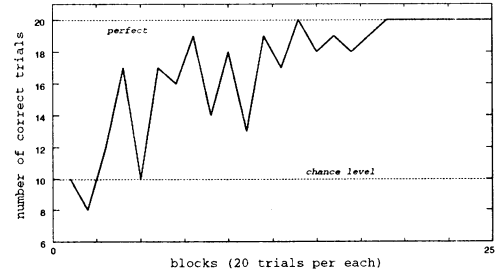
Figure 4: The rate of correct response with learning blocks

Next, figure (5) gives the overlayed raster plot of two trials. In the early stage of learning(fig. 5 top), the dynamics is not yet self organized and there is quite a few mismatches of neuronal firing with the same condition. We can see that output is also different for the two trials. On the other hand, in the late stage of learning (fig. 5 below), there are only few mismatches of neuronal firing for the two trials. As a result, the output for the two trials are both correct.

Finally, we show the increased synaptic connection due to the two learning rules after the pattern input(fig. 6). First half of the increased connections are due to the temporal hebbian rule, which self organizes the dynamics. The second half

which connects the self organized dynamics with the desired output is due to the reinforcement learning.
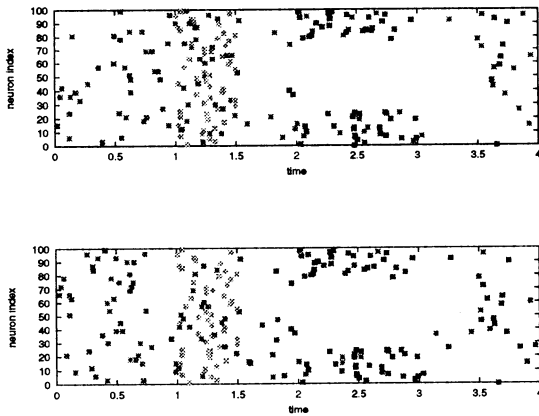
Figure 5: Two trials overlayed for early (top) and late (below) stage of learning
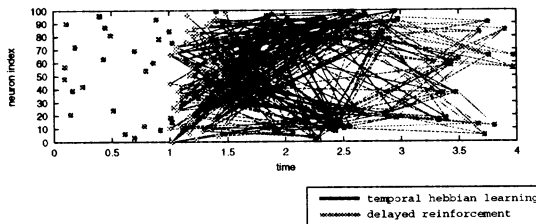
Figure 6: Increased synaptic connections due to temporal hebbian learning and reinforcement learning

5. Discussions

We have proposed a network model which utilizes functional connectivity and realizes the switching of pattern response by context. Due to functional connectivity, the basic network dynamics changes according to context which is given as a spatio-temporal spike pattern. Then the two learning rules functions on the basic dynamics to coordinate the pattern input and the correct response. The temporal hebbian rule works to self organize the dynamics so that a nearly identical firing pattern appears when given the same set of context and pattern input. The role of reinforcement learning is to couple the self organized dynamics with the correct output by trial and error.

References

[1] Robinson, D.A. (1975) Oculomotor control signals. In P. Bach-y-Rita, G. Lennerstrand (Eds.) Basic mechanisms of ocular mobility and their clinical implications. Oxford: Pergamon Press, vol.23, pp.337–374.

[2] Von der Malsburg, C. (1987) Synaptic plasticity as basis of brain organization. In:Changeux JP, Konishi M (ed) The neural and molecular basis of learning, John Wiley and Sons, New York.

[3] Softky, W. R., Koch, C. (1993) The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. J. Neurosci, 13, 334–350.

[4] Shadlen, M.N., Newsome, W.T. (1994) Noise neural codes and cortical organization. Current Opinion in Neurobiology, 4, 569–579.

[5] Softky, W.R. (1994) Sub-millisecond coincidence detection in active dendritic trees. Neurosci., 58, 13–41.

[6] Vaadia, E., Haalman, I., Abeles, M., Bergman, H., Prut, Y., Slovin, H., Aertsen, A. (1995) Dynamics of neuronal interactions in monkey cortex in relation to behavioral events. Nature, 373, 515–518.

[7] Sakurai, Y. (1993) Dependence of functional synaptic connections of hippocampal and neocortical neurons on types of memory. Neuroscience Letters, 158, 181–184.

[8] Tsuda, I. (1992) Dynamic link of memory - chaotic memory map in nonequilibrium neural networks. Neural Networks, 5, 313–326.

[9] Gerstner, W., Kempter, R., Hemmen, J.L.V., Wagner, H. (1996) A neuronal learning rule for sub-millisecond temporal coding. Nature, 383, 76–78.

[10] Markram, H., Lübke, J., Frotscher, M., Sakmann, B. (1997) Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. Science, 275, 214–215.

[11] Watanabe M., Aihara K.(1997) Chaos in neural networks composed of coincidence detector neurons, Neural Networks, 10:8:1353–1359.

[12] Watanabe M., Aihara K. and Kondo S. (1998) A dynamical neural network with temporal coding and functional connectivity,Biological Cybernetics, 78:87–93.

[13] Fujii, H., Aihara, K., Hamada, K. (1995) A dynamical cell assembly model by mutual correlation - towards spatio-temporal coding by a system of coincidence detecting neurons. In: Proc. international Symposium on Nonlinear Theory and its Applications(NOLTA '95) ,pp 109–114.

A Simplified Single Neuron Model as a Rapid Depolarization Detector Based on the H-H Type Neuron

Yuichi Sakumura [*], Kazuyuki Aihara [*,†]

[*] Dept. of Complexity Science and Engineering, The University of Tokyo,
[†] Dept. of Mathematical Engineering and Information Physics, The University of Tokyo
Tokyo 113–8656, Japan
e-mail: {saku,aihara}@sat.t.u-tokyo.ac.jp

Abstract

Real neuron may generate spikes when the membrane potential is rapidly depolarized by synaptic inputs rather than when the membrane potential exceeds a certain level of the potential. The Hodgkin-Huxley type neuron also shows the sensitivity to the rapid depolarization of the membrane potential. In this paper, we analyze the relation between the input and the output spike timings of the Hodgkin-Huxley type neuron and propose the simplified neuron model which has similarity with the Hodgkin-Huxley neuron in their output spike patterns.

1 Introduction

When we construct a large network compounded of single neurons, it is important to simplify the mechanism of the neuron. For this purpose, the functions and the properties of a realistic neuron should be well studied. From the physiological point of view, the neuronal sensitivity depends on more than input frequency [9–11], in other words, a rapid depolarization provokes an action potential [12]. Such property is explained by the kinetics of sodium channels [13]. This fact shows that physiological neurons differ from theoretical single neuron models which do not detect the increase of membrane potential but the amplitude itself.

There are many theoretical single neuron models which are developed from McCulloch-Pitts neuron [1]. The leaky integrate-and-fire neuron

[2], one of such models, is used in order to solve the question of how and why biological neurons fire irregularly [3–5]. This question is also studied by using the more complex neuron model which has the dynamical mechanisms of some kind of ion channels on the membrane [6–8]. Though the most important point to study this question is the complex mechanism of spike generation, the leaky integrate-and-fire neuron model may neglect it, namely, the voltage threshold should not be a fixed value because the threshold of a real neuron is variable.

In this paper we examine the quantitative measure of how the Hodgkin-Huxley type neuron (H-H neuron) [13], one of more realistic single neuron models, is sensitive to rapid depolarization. By utilizing this measure, we construct the much simplified model and compare the firing time of the simplified model with that of the H-H neuron. Our results indicate that our neuron model as a rapid depolarization detector is a good simplification of the H-H neuron in the firing time.

2 Analysis of the H-H neuron

First, to analyze the input-output relation of the H-H neuron, we compute the following Hodgkin-Huxley equations,

$$C_m \frac{dV}{dt} = -I_{Na} - I_K - I_l - I_{syn} \qquad (1)$$

where V (mV) is the membrane potential, and $C_m (= 1.0 \mu F/cm^2)$ is the membrane capacitance
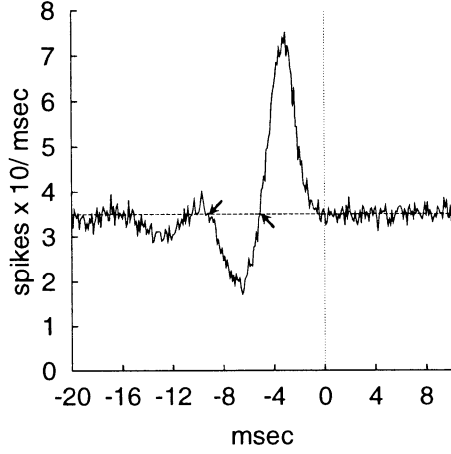
Figure 1: Reverse correlation of excitatory synaptic inputs with the bin width 0.1 msec. The frequency is 3500 Hz. The vertical axis represents the number of excitatory inputs per each output spike which has no other preceding spikes within at least 20 msec. Two arrows point to the intersections of the reverse correlation curve and the mean (3.5 spies/msec).

Figure 2: Parameters of the simplified model.

per unit area. The ionic currents I_{Na} (sodium), I_K (potassium), and I_l (passive leak) are defined as $I_{Na} = 120m^3h(V - V_{Na})$, $I_K = 36n^4(V - V_K)$ and $I_l = 0.3(V - V_l)$, where the reversal potentials are $V_{Na} = 50.0 - E_{rest}$, $V_K = -77.0 - E_{rest}$ and $V_l = -54.4 - E_{rest}$, respectively (the resting potential is $E_{rest} = -65$ mV). The gating variables m, h, and n obey a kinetic equation $dx/dt = \{\alpha_x \cdot (1 - x) - \beta_x \cdot x\} 3^{(15-6.3)/10}$, where α_x and β_x ($x = m, h, n$) are $\alpha_m = 0.1(25-V)/\{e^{(25-V)/10} - 1\}$, $\beta_m = 4e^{-V/18}$, $\alpha_n = 0.01(10 - V)/\{e^{(10-V)/10} - 1\}$, $\beta_n = 0.125e^{-V/80}$, $\alpha_h = 0.07e^{-V/20}$ and $\beta_h = 1/\{e^{(30-V)/10} + 1\}$. The neuron expressed by equation (1) has $V = 0.0$ (mV) at the resting condition. The term I_{syn} represents the summation of all synaptic currents where $I_{syn} = (\sum_i g_{syn}(t - t_i)) \cdot (V - V_{syn})$ and t_i is the instant of the i-th synaptic input. We suppose $g_{syn}(t)$ as the fast excitatory AMPA ($V_{syn} = -10 - E_{rest}$ mV) types of synaptic conductances per unit area defined by the alpha func-

tion [14, 15] $g_{syn}(t) = Ate^{1-t/t_{peak}} (t \geq 0)$ where $A = g_{peak}/4\pi r^2 t_{peak}$ and g_{peak} (nS) is the peak value of the synaptic conductance at the peak time t_{peak} (msec); $g_{peak} = 0.5$ and $t_{peak} = 2.0$. For the sake of simplicity, we suppose that the neuron is composed of a single compartment of a sphere shape with radius $r = 20$ μm, ignoring the complicated morphology of dendrites.

The "reverse correlation" is the histogram of synaptic input times around firing times in which each input time is measured from the corresponding firing time. Figure 1 shows the reverse correlation of H-H type neuron expressed by Eq. (1) when t_i is produced by Poisson process of the frequency 3500 Hz. This figure shows that the neuron fires when there are an immediate sharp *peak* just after a sudden *dip* of the excitatory inputs. Such fluctuation is observed in the physiological experiment [12], not but in the theoretical study with a leaky integrate-and-fire neuron [8].

3 Simplified model

To construct a simplified neuron model based on the H-H neuron, we measure here the "time width" and the "area" of the *peak* (T_p and S_p) and *dip* (T_d and S_d) as shown in Fig. 2. We mean the term "time width" as the time width within which the number of spikes is more or less than the mean, and the term "area" as its total

difference (see Fig 2). From Fig. 1, we obtaines $T_p = 5.0$ msec, $S_p = 8.45$ spikes, $T_d = 4.1$ msec and $S_d = -3.92$ spikes.

Our simplified model just counts the number of inputs within consecutive two time bins, T_p and T_d. The neuron fires at time t if the number of input spikes in $[t - T, t]$, $N[t - T, t]$, satisfies the following conditions:

$$N[t - T_p - T_d, t - T_p] \leq S_d \qquad (2)$$
$$N[t - T_p, t] \geq S_p. \qquad (3)$$

For the simplicity, we do not consider the effect of the refractory period.

4   Results

The results of the input frequency of 3500 Hz are shown in Fig. 3. They are obtained by computing Eq. (1) (solid line) and just counting the input number with $T_p = 5.0$, $S_p = 8.45$, $T_d = 4.1$ and $S_d = -3.92$ (black circle). These figures shows that our simplified neuron model often fires at the same time with the H-H neuron.

Though examples illustrated in Fig. 3 shows only the case of 3500Hz of excitatory inputs and no inhibition, we can construct neuron model based on the H-H neuron under any other input conditions since the modeling parameters $(T_p, S_p, T_d$ and $S_d)$ are calculated in the same way. This implies that the neuronal mechanism changes dynamically depending on the input condition because the dynamics of the membrane is changed by the total conductance on the membrane $(-\sum_i g_i, i = Na, K, l, syn)$. Our model also contributes the variability to neuronal firing patterns. Indeed, the coefficients of variation of the interspike intervals in Fig. 3 are 0.447, 0.614, 0.813, 0.769, 0.637, 0.663 from the top.

The Fitzhugh-Nagumo equations [16, 17] were proposed as the reduced H-H equations from the qualitative approach and well modeled in the qualitative property of the membrane. It is, however, different from the H-H model in the scale of variables ("time" for example). Supposing that only timings of spikes carry neuronal information,
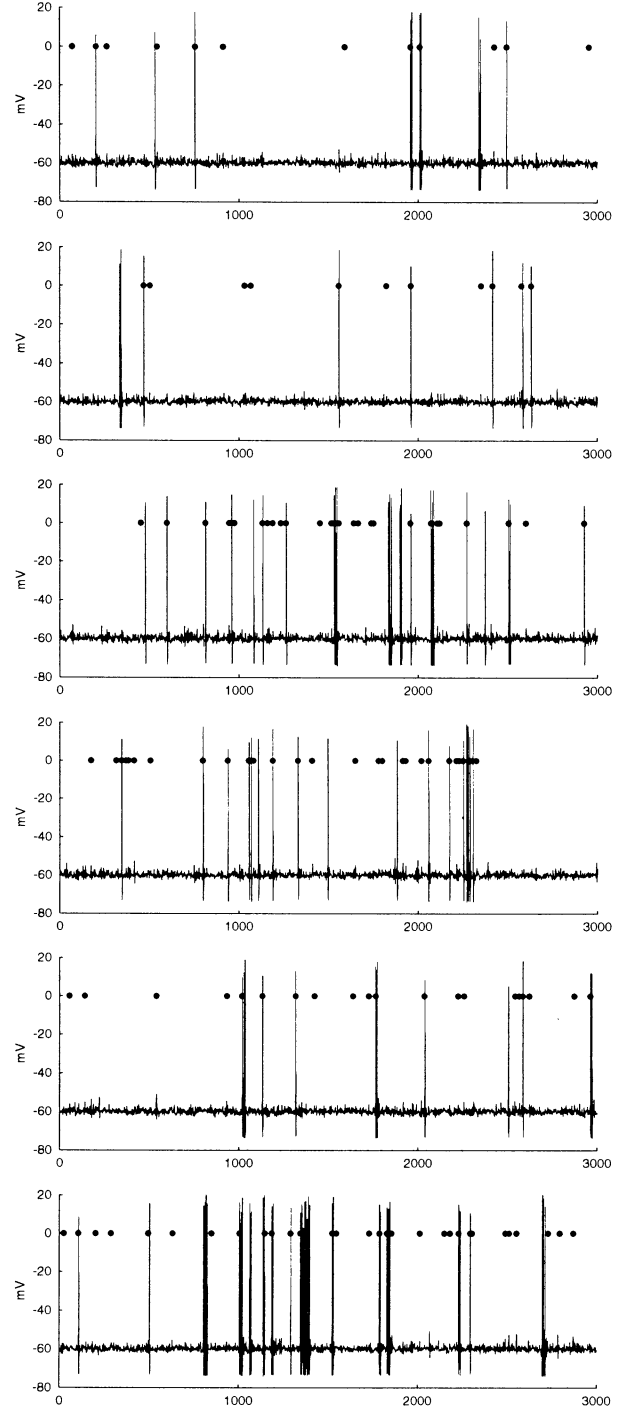
Figure 3: Comparisons of the firing times of the H-H neuron (solid line) and the simplified model (black circle). Good estimations of firing times at low firing rate (the top two figures), at high firing rate (the middle two ones) and worse estimations (the bottom two ones).

our model that is made in quantitative way may be enough to simulate the information processing. It remains to be proved how we improve our model to fit the firing pattern to the H-H model more precisely.

5 Conclusion

We shows that the firing times of our very simplified neuron model as a depolarization detector are very similar to those of the H-H neuron models. If the model is improved to fire more precisely, we can construct very large H-H type network and simulate it very fast without using the complex H-H equations.

Acknowledgements

References

[1] W. S. McCulloch and W. H. Pitts. A logical calculus of the ideas immanent in neural nets. *Bull. Math. Biophys.*, 5:115–133, 1943.

[2] R. Stein. Some models of neuronal variability. *Biophys. J.*, 7:37–68, 1967.

[3] M. N. Shadlen and W. T. Newsome. Noise, neural codes and cortical organization. *Curr. Opin. Neurobiol.*, 4:569–579, 1994.

[4] M. N. Shadlen and W. T. Newsome. Is there a signal in the noise? *Curr. Opin. Neurobiol.*, 5:248–250, 1995.

[5] S. Shinomoto, Y. Sakai, and S. Fukahashi. The ornstein-uhlenbeck process does not reproduce spiking statistics of neurons in prefrontal cortex. *Neural Computation*, 11:935–951, 1999.

[6] W. R. Softky and C. Koch. The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs. *J. Neuroscience*, 13(1):334–350, 1993.

[7] W. R. Softky. Sub-millisecond coincidence detection in active dendritic trees. *Neuroscience*, 58(1):13–41, 1994.

[8] W. R. Softky. Simple codes versus efficient codes. *Curr. Opin. Neurobiol.*, 5:239–247, 1995.

[9] H. Markram and M. Tsodyks. Redistribution of synaptic efficacy between neocortical pyramidal neurons. *Nature*, 382:807–810, 1996.

[10] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275:213–215, 1997.

[11] A. M. Thomson. More than just frequency detectors? *Science*, 275:179–180, 1997.

[12] Z. F. Mainen and T. J. Sejnowski. Reliability of spike timing in neocortical neurons. *Science*, 268:1503–1506, 1995.

[13] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544, 1952.

[14] W. Rall. Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic inputs. *J. Neurophysiol.*, 30:1138–1168, 1967.

[15] J. J. B. Jack, D. Noble, and R. W. Tsein. *Electrical Current Flow in Excitable Cells.* Claredon Press, Oxford, second edition, 1975.

[16] R. FitzHugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophys. J.*, 1:445–446, 1961.

[17] J. Nagumo, S. Arimoto, and S. Yoshizawa. An active pulse transmission line simulating nerve axon. *Proc. IRE.*, 50:2061–2070, 1962.

Effect of small random noise on the chaotic behavior of the Hodgikin-Huxley equations

Hiroaki Tanaka

Kazuyuki Aihara
Department of Mathematical Engineering
and Information Physics
The Univ. of Tokyo
Bunkyo-ku, 113-8656, Japan

CREST, Japan Science and Technology Co. (JST)
4-1-8 Hon-cho, Kawaguchi, 332, Jaoan

Abstract

We found previously that the chaotic behavior of the Hodgikin-Huxley equations is modulated by small white noise[1]. In this study we try to evaluate the behavior quantitatively. The results confirm that amplitude dependence of the noise effect on the chaotic behavior is caused by two different mechanisms.

Key words: Hodgikin-Huxley equations, noise, order, information entropy

Introduction

The excitable membrane potential of the squid giant axon shows chaotic behavior under certain sinusoidal current stimulation[2,3,4]. In the actual neuron or the physiological experiment using giant axon, neuron is exposed to the noise of certain amount of thermal and/or electrical fluctuation (i.e. noise).

Matsumoto and Tsuda[5] showed that a certain amount of noise induced some order in the B-Z chaos (NIO). Thus, we supposed that the observed chaos in the neuron may be affected by the noise. The chaos without noise in the neuron is well simulated by the numerical calculation of Hodkin-Huxley equations[6]. In this paper, the effects of the noise with various amplitudes on neuron are simulated numerically with the Hodgkin-Huxley equations.

Simulation

Detail methods and parametric conditions for the numerical calculation are based on Usami et.al.[6]. White noise is superimposed to the sinusoidal current stimulation.

We have studied the following points: (1)wave form of the membrane potential, (2)interval of the peaks and its standard deviation, (3)Poincare sections, (4)peak height distribution, (5)return maps of the peaks and its sequence, (6)information entropy with various bin size and (7)frequency of the orbital deviation.

Results

(1)Wave form

By comparing the chaotic wave forms of the membrane potential with noise of various amplitudes, it was found that the wave form is changed by the noise. Obvious difference in the wave forms was observed with noise larger than 1. The similar sequence of peaks with almost same interval was frequently observed, suggesting some macroscopic order. This seems to be accompanied with the shortening of the interval.

(2)Interval and standard deviation

It was found that the interval of the peaks of the waveform is decreased about seceral %, and the standard deviation decreases about 40%, when the noise amplitude is larger than 1 (Fig.1).
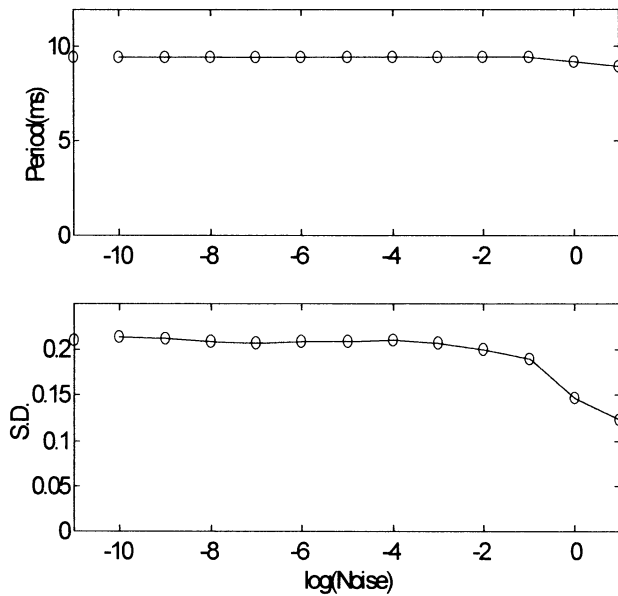
Fig.1 Intervals of the peaks and those standard deviations. They are plotted with changing the amplitude of noise. The points on the Y-axis indicate the values without noise.

(3)Poincare section

It was found that fine structure is observed in Poincare section, which is observed with certain phase of the sinusoidal current stimulation. When the noise is increased, the points on the section are fluctuated and the structure is smeared. The figure of the Poincare section shows certain critical point, which may sensitive to the noise to deviate the trajectory.

(4)Peak heights

The distribution of the peak heights of the wave forms of the membrane potential is analyzed. Without noise, five major peaks and 11 minor peaks are observed in the peak height distribution. At the small noise amplitude, although the wave form can not be distinguished from that without noise and the major peaks are not affected, although the 11 minor peaks in the distribution increase more than twice than those without noise. The increase of the 11 peaks is observed at the noise amplitude up to 10^{-3}. This increase of the peaks is caused by some convergence of the trajectories.

Above 10^{-2} to 10^{-1}, the 11 minor peaks decrease to vanish, while some of the major peaks increase. This seems that the trajectories are diverged.

Above 1, all the peaks become smaller with broadening peak height distribution, suggesting whole divergence in the trajectory is caused by large noise.

(5)Return map and peak sequence

From the peaks of the wave forms, return maps are obtained for the various noise amplitudes. It was found that fundamental structure of the return map is not changes by the noise.

The heights of five major peaks and the 11 minor peaks are consistent with each other on the return map. In addition, the increase of the 11 minor peaks can be explained by the increase of the points located at the specific region on the diagonal of the map. The points of the region increase about 12% than without noise, while they decrease significantly with the increasing noise amplitude.

At larger noise, the points relevant to the macroscopic periodicity increase, while some other points decrease. The peak sequence analysis also shows the increase of short term wandering on the map.

(6)Information entropy

The information entropy is calculated with various size of bin. It was found that at the small bin size the information entropy increases with the noise, while at larger bin it slightly decreases until the noise is less than 10^{-1}. It is related to NIO (Fig.2).
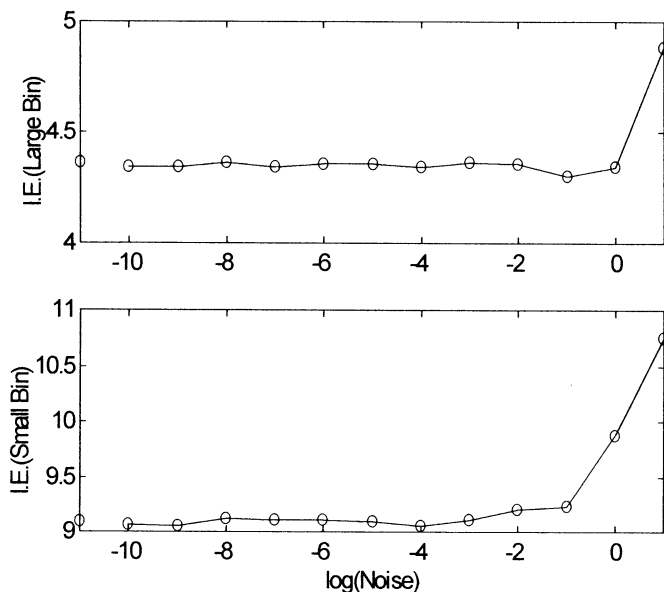
Fig.2. Information entropy of the Poincare section. They are plotted with changing the amplitude of noise. The points on the Y-axis indicate the values without noise. Upper and lower figures show the case of larger bin (about 0.03) and smaller bin (about 0.001), respectively.
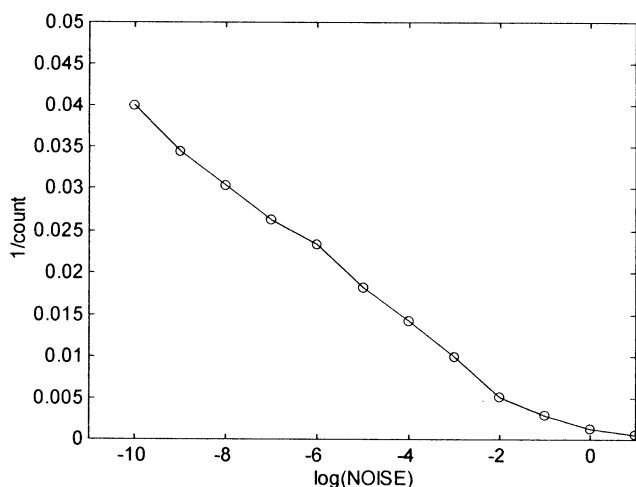
Fig.3. The inverse of the frequency of the major trajectory deviation is plotted. The deviation points are counted for 5000 cycles of the external sinusoidal current.

(7)Noise induced trajectory deviation

To examine the effect of the noise in detail, the time courses of the trajectories with some noise and without noise were calculated simultaneously with checking the distance between those in the phase space. It was found that initially both time courses were in the same manner, but became independently. Thus, the number of the deviation within the same time duration is counted for the various noise amplitudes (Fig.3).

It was found that inverse of the frequency is proportional to the log scaled noise, and its relation shows bi-phasic manner. Theoretically, the relation should follow the simple relation.

1/frequency = const − α log(noise size)

When the noise size is small, α is almost 20, while if the noise is larger than 0.1, it becomes 4. This exhibits the existence of two types of the mechanism which affect the chaotic behavior.

Discussion

In this study, we have applied various methods to evaluate the chaotic behavior quantitatively. The convergence of the trajectory and decrease of information entropy are observed. These may be related to the noise-induced order (NIO) previously reported by Matsumoto and Tsuda[5]. They discussed that the NIO is determined by balance of the trajectory convergence and divergence induced by the trajectory transition with noise. The results of our

study with smaller noise indicated the contribution of the NIO.

On the other hand, another mechanism is involved in this chaos, which is observed as the macroscopic behavior. The critical point in the Poincare section may be related to this order.

References

[1] Tanaka H, Aihara K (1997), Chaotic behavior of the Hodgikin-Huxley equations under small random noise. Complexity and Diversity. Springer-Verlag, Tokyo, pp.172-174

[2] Aihara K, Matsumoto G, Ikegaya Y (1984), Periodic and non-periodic responses of a periodically forced Hodgkin-Huxley oscillator. J. theor. Biol. 109:249-269

[3] Aihara K, Matsumoto G, Ichikawa M (1985), An alternating periodic-chaotic sequence observed in neural oscillators. Phys. Lett. A 111:251-255

[4] Aihara K, Numajiri T, Matsumoto G, Kotani M (1986), Structures of attractors in periodically forced neural oscillators. Phys. Lett. A 116:313-317

[5] Matsumoto K, Tsuda I (1983), Noise-induced order. J. Stat. Phys. 31:87-106

[6] Usami T, Yamada T, Ichinose N, Aihara K(1995), Hodgikin-Huxley equation and its response to the periodic stimulation. J.SICE, 34:769-774

A Mixed Analog/Digital Circuit for Quadratic Assignment Problems

Hiroshi Hayashi and Yoshihiko Horio
Department of Electronic Engineering
College of Engineering
Tokyo Denki University
Chiyoda-ku, Tokyo 101-8457, Japan

Kazuyuki Aihara
Department of Mathematical Engineering
and Information Physics
The University of Tokyo
Bunkyo-ku, Tokyo 113, Japan

Abstract

In this paper, a mixed analog/digital circuit for quadratic assignment problems, using chaotic neural network, is presented. A part of neuron in this proposed system is constructed by extending a current-mode chaotic neuron circuit. Furthermore, we discuss about a modification of an original method to suit the hardware system.

Keywords: A Mixed Analog/Digital Circuit, Chaotic Neuron, Quadratic Assignment Problems

1 Introduction

In recent years, chaotic phenomena have been observed in various systems. Furthermore, chaos and its applications have been investigated by many researchers. As a one of its application, applying chaotic dynamics to combinatorial optimization problems has been received great attentions.

Combinatorial optimization problems have various types and there are many problems which have not been found optimum solutions yet. A great number of methods to solve such problems have been proposed.

For example, there are the methods based on the conventional Hopfield neural network with chaotic dynamics. In this method, chaotic dynamics was used in order to avoid a local optimum, and it have been shown that they have a very high ability to solve the problems.

On the other hand, many strong heuristic methods have been proposed for combinatorial optimization problems.

A new approach which combines chaotic dynamics and one of heuristic methods called the tabu search has been proposed for quadratic assignment problems[1][2][3]. The tabu search is the method which stores the states which have already searched

as a tabu list and avoids to return the states listed on the tabu list. In this approach, chaotic neural network model[5] has been used.

On the other hand our group has also proposed some chaotic neuron circuits[6],[7],[8],[9],[10],[11],[12] using various type of circuit techniques. In this paper, as an extension of the current mode chaotic neuron circuit[8], we propose the current mode chaotic neuron circuit for quadratic assignment problems.

Our final purpose is constructing the mixed analog/digital circuit for quadratic assignment problems realizing a new approach which is mentioned in the above. However, in the case that we construct the hardware system, we need to modify the approach at some points for the hardware implementation. In our simulation we have confirmed that modified approach has almost same solving ability as that of the original one. By constructing hardware system to solve the quadratic assignment problem, a shorter computing time than computer simulation can be realized.

2 Quadratic Assignment Problems

Quadratic assignment problem(QAP) is one of the NP-hard combinatorial optimization problems and defined as follows: when distance matrix A and flow matrix B are given, find the permutation p which give the minimum objective function value. The objective function is

$$f(p) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{p(i)p(j)}, \qquad (1)$$

where a_{ij} and b_{ij} are the (i, j) th element of matrix A and B, respectively. $p(i)$ is the i th vector p, and n is the size of the problem.

3 Chaotic Neural Network Realizing the tabu search Effects

Utilizing the tabu search realized by neural network is similar to the chaotic neural network model[5], therefore, a new approach for quadratic assignment problems is proposed[1][2][3]. The chaotic neuron which constitutes the network for the new approach is expressed as

$$\xi_{ij}(t+1) = \beta(D_{current}(t) - D_{ij}(t)) \qquad (2)$$

$$\eta_{ij}(t+1) = -W\sum_{k=1}^{n}\sum_{l=1}^{n} x_{kl}(t) + k_r\eta_{p(j)q(i)}(t) \\ -\alpha(x_{p(j)q(i)}(t) + z_{p(j)q(i)}(t)) + R \quad (3)$$

$$\zeta_{ij}(t+1) = k_r\zeta_{ij}(t) - \alpha(x_{ij}(t) + z_{ij}(t)) + R \quad (4)$$

where $\xi_{ij}(t+1)$, $\eta_{ij}(t+1)$ and $\zeta_{ij}(t+1)$ are the internal states of the (i,j) th neuron at the discrete time $(t+1)$ corresponding to the gain effect, feedback inputs from other neuron in the network and refractoriness of the $(p(j)q(i))$ th exchange, and refractoriness of the (i,j) th exchange, respectively, $D_{current}(t)$ is the current minimum objective function values of the permutation p at time t, $D_{ij(t)}$ is the new objective function values of the permutation that is made by exchanging j th and $q(t)$ th elements of p at time t, β is the scaling parameter for the gain effects, W is the connection weight between the (i,j) th neuron and other neurons, k_r is the decay parameter of the tabu effects, α is the scaling parameter of the tabu effects, R is the strength of the external input.

Furthermore, the total internal state of the (i,j) th neuron at the discrete time $(t+1)$, $y_{ij}(t+1)$, and its output, $x_{ij}(t+1)$, are described by

$$y_{ij}(t+1) = \xi_{ij}(t+1) + \eta_{ij}(t+1) + \zeta_{ij}(t+1) \quad (5)$$

$$x_{ij}(t+1) = f(y_{ij}(t+1)) \qquad (6)$$

where $f()$ is the nonlinear output function of the neuron.

When $x_{ij}(t+1)$ is larger than 0, the (i,j) th neuron fires. Then the element i is placed to the j th index, at the same time the element $p(j)$ is placed to the $q(i)$ th index, respectively. Furthermore, $z_{ij}(t+1)$ is reset to 0 and $x_{ij}(t+1)$ is added to $z_{p(j)q(i)}(t+1)$. The $x_{ij}(t+1)$ which is large enough to fire prevent to replace element i to j th index, in the same way as that one the large $z_{p(j)q(i)}(t+1)$ prevent to replace element $p(j)$ to $q(i)$ th index.

4 Extended Current-Mode Chaotic Neuron Circuit for Quadratic Assignment Problems

4.1 Current-Mode Chaotic Neuron Circuit

The chaotic neuron model with one internal state is described by

$$y(t+1) = ky(t) - \alpha x(t) + a \qquad (7)$$

$$x(t+1) = f(y(t+1)) \qquad (8)$$

and a block diagram representing equations (7) and (8) is shown in Fig. 1.
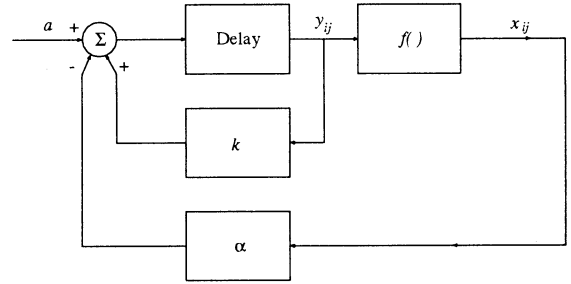
Figure 1: A block diagram of chaotic neuron model

Fig. 2 shows the current-mode chaotic neuron circuit which our group has already implemented. This circuit mainly consists of CMOS inverters, which are used as transconductance amplifiers and nonlinear element.

4.2 Extended Circuit for Quadratic Assignment Problems

Extending the current-mode chaotic neuron circuit, we propose the neuron circuit for quadratic assignment problems. The extended block diagram representing equations (2), (3), (4), (5) and (6) is shown in Fig. 3.

The extended circuit which we propose is shown in Fig. 4, where part of $\eta_{ij}(t + 1)$ is gathered for simplification.

5 Simulations

For realizing a new method as a hardware system, it needs modifications to be able to apply to a hardware system. Because of doing modifications, there is a possibility of large declining in its solving ability. We have investigated the effect of modifications.
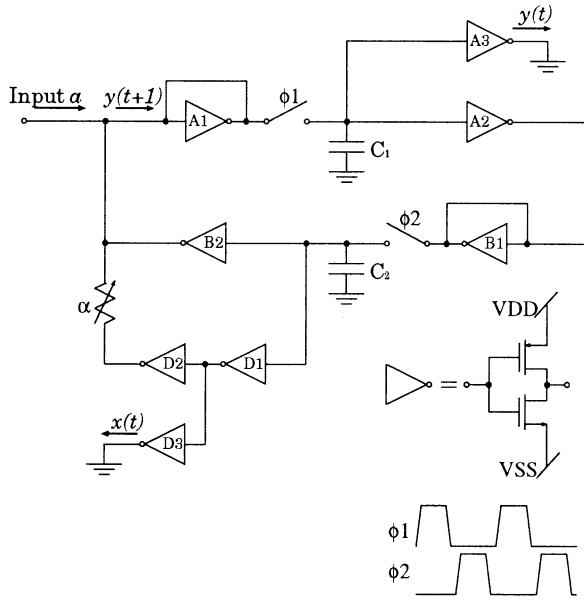
Figure 2: A schematic diagram of current mode chaotic neuron circuit

Figure 3: A block diagram of the extending circuit for quadratic assignment problems

Figure 4: A schematic diagram of extending current mode chaotic neuron circuit for quadratic assignment problems

One modification point is about a nonlinear output function part. In a based method, a sigmoidal function is used as a nonlinear output function. However, the nonlinear output function circuit in the current-mode chaotic neuron circuit is based on a simple inverter's nonlinearity, and the characteristics between the former and the latter is different. We use the circuit characteristics for simulation.

Second is about the value transmitted from one neuron to other neurons. In a based method, all of such kind of values are continuous. Since it is difficult that all mutual transmitted values are realized by continuous values, we change these values to digital values.

Third is about the limitation of the values. Although there are no limitation in a based method, we need to consider values' limitation for the circuit characteristics.

We have compared the solving ability from a original method and a modified method which included these modification, and have gotten almost same

Table 1: The result in the case of changing resolution of the gain effect

A type of method	A probability of the arrival at the best known solution			
	without modifications	with modifications		
$\beta(D_{current}(t) - D_{ij}(t))$	continue	continue	8bit resolution	4bit resolution
Tail12	100%	100%	100%	100%
Tail15	100%	84.5%	82.7%	81.2%
Tail17	100%	75.4%	74.3%	72.7%

solving ability. Here, we have especially investigated effect of the precision of the gain effect values, $\beta(D_{current}(t) - D_{ij}(t))$. Table 1 shows the simulation result. Although there are declines in the method with modifications against the based one without modifications, the hardware system which has shorter computing time executes much more iterations among same computing times. This advantage of the hardware is enough to realize shorter computing time to arrive the best known and optimum solution.

6 Conclusion

In this paper, we proposed current-mode chaotic neuron circuit for quadratic assignment problems, extending simple one. Then we investigated the effect of modification. Moreover, to construct a mixed analog/digital circuit system for quadratic assignment problems, a circuit for computing objective function values and a whole structure of the system are under designing.

References

[1] M. Hasegawa, T. Ikeguchi and K. Aihara, "A Novel Chaotic Search for Combinatiorial Optimization", in *Proc. NOLTA'97*, pp. 613-616, 1997.

[2] M. Hasegawa, T. Ikeguchi and K. Aihara, "A New Modern Heuristic Approach using Chaotic Dynamics for Quadratic Assignment Problems", in *Tech. Rept.of IEICE, NLP97-8*, pp. 53-60, 1997.

[3] M. Hasegawa, T. Ikeguchi and K. Aihara, "On the Relation between Chaotic Neural Network Approach and Tabu Search Approach for Combinatorial Optimization", in *Tech. Rept.of IEICE, NLP97-49*, pp. 73-80, 1997.

[4] M. Hasegawa, T. Ikeguchi and K. Aihara, "Solving Quadratic Assignment problems by Chaotic Neural Networks", in *Proc. ICONIP'97*, pp. 182-185, 1997.

[5] K. Aihara, "Chaotic neural networks", in *Bifur. Pheno. in Nonlinear Syst. and Theory of Dynamical Syst.*, H. Kawakami ed., pp. 143-161, World Scientific, 1990.

[6] Y. Horio and K. Suyama, "IC Implementation of Switched-Capacitor Chaotic Neuron", in *Proc. IEEE Int. Symp. Circuits Syst.*, Vol. 6 of 6, pp. 97-100, 1994.

[7] Y. Horio and K. Suyama, "IC implementation of switched-capacitor chaotic neuron", in *Proc. IEEE ISCAS'95*, pp. 97-100, 1995.

[8] R. Herrera, K. Suyama, and Y. Horio, "IC implementation of a current-mode chaotic neuron", in *Proc. IEEE ISCAS'98*, vol. 3, pp. 546-549, 1998.

[9] N. Kanou, Y. Horio, K. Aihara and S. Nakamura, "A current-mode circuit of a chaotic neuron model", in *IEICE Trans. on Fundamentals*, vol. E76-A, no. 4, pp. 642-644, 1993.

[10] N. Kanou, Y. Horio, K. Aihara and S. Nakamura, "A Current-mode circuit of a chaotic neuron model using SI ntegrator", in *IEICE Trans. on Fundamentals*, vol. E77-A, no. 1, pp. 335-338, 1994.

[11] H. Hayashi, Y. Horio and K. Aihara, "Switched-capacitor chaotic neuron circuit with three internal states", in *Proc. NOLTA'98*, pp. 479-482, 1998.

[12] I. Kobayashi, H. Hayashi, M. Yoneda and Y. Horio, "Nonlinear Output function Circuits for SC Chaotic Neuron Circuit", in *Proc. of the 1998 IEICE General Conference*, Vol. 1, p. 26, 1998.

A Switched-Capacitor Chaotic Neuro-Computer : Medium-Scale Implementation

Masaaki Yoneda, Izumi Kobayasi,
Ou Yamamoto and Yoshihiko Horio
Department of Electronic Engineering
College of Engineering
Tokyo Denki University
Chiyoda-ku, Tokyo 101-8457, Japan

Kazuyuki Aihara
Department of Mathematical Engineering
and Information Physics
The University of Tokyo
Bunkyo-ku, Tokyo 113, Japan

Abstract

We are building a very large scale chaotic neural network with analog chaotic neuron. The computer is composed of 10,000 chaotic neurons and 100,000,000 synapses. In this paper, a specification of the network system and a method to compose the neural networks is introduced.

Keywords: Chaos, Neural Networks, VLSI

1 Introduction

A chaos phenomenon has been adopted very much in the neural network system since it is discovered that there is chaos in the condition of the information transmission of the brain. The chaotic neuron model [1] is proposed as one of the neural network model. The chaotic neural network system is applied to the various applications such as difficult problem to solve in the usual method. For example, it is used the traveling salesman problems (TSP), the quadratic assignment problems (QAP) [2, 3] and the dynamical associative memory (DAM) [4].

However, when a large-scale problem is solved with the usual computer even if the method that described the above is used, it takes time very much. Therefore, we are making the large-scale network system that realized with the circuit.

The chaotic neuron circuit with single-internal-state proposed [2, 3]. However, these circuits are restricted to a result for single-state. Therefore, the neuron circuit that we used in the network system is switched capacitor (SC) chaotic neuron circuit with three-internal-state [7, 8]. A network system that introduce in this paper is connected with the three-state chaotic neuron. And a synapse for connective interface

of theses neurons is presented [9].

In this paper, we introduce these neuron and synapse circuit, and propose specification and structure of chaotic neural network system to satisfy the introduced model [1]. This chaotic neural network system is used 10,000 neurons and 100,000,000 synapses.

2 Chaotic Neuron Model

The model that we used this time was introduced as chaotic neuron model with three internal state [1]. The output x_i of i th neuron can be shown as follow

$$x_i(t+1) = f(y_i(t+1)) \quad (1)$$

where, $f(\cdot)$ is the nonlinear output function, and y_i is the internal state of the i th neuron. And the internal state of the neuron can be composed by the parts of three that written by

$$
\begin{aligned}
y_i(t+1) = &\underbrace{k_e \xi_i(t) + \sum_{j=0}^{m} v_{ij} A_j(t)}_{\xi_i(t+1)} \\
&+ \underbrace{k_r \zeta_i(t) - \alpha x_j - \theta_i}_{\zeta_i(t+1)} \\
&+ \underbrace{k_f \eta_i(t) + \sum_{j=0}^{n} w_{ij} x_j(t)}_{\eta_i(t+1)} \quad (2)
\end{aligned}
$$

The first part, $\xi_i(t+1)$ is the internal state about External input. The state has decay parameter of external input, K_e, the connective weight from j th external input to i th neuron, v_{ij}, and j th external input strength, A_j.

The second part, $\zeta_i(t+1)$ is the internal state about relative refractoriness. This state has self-refractoriness parameter, k_r, the refractoriness parameter, α.

And the third part, $\eta_i(t+1)$ is the internal state about input of feedback from neurons which compose a network. Where, k_f is parameter of feedback from other neurons, n is the total number of neurons in network, w_{ij} is connective weight from j th neuron to i th neuron.

3 Used Circuits In The System

We thought that internal state in Eq. (2) was divided, as the part of chaotic neuron with analog technology and the part of synapse with digital technology, for chaotic-neuro computer system. It is shown by Eq. (3),(4).

$$y_i(t+1) \;=\; \underbrace{\xi_i(t+1) + \zeta_i(t+1) + k_f\eta(t)}_{neuron\ part} \quad (3)$$

$$+ \underbrace{\sum_{j=0}^{n} w_{ij}x_j(t)}_{synapse\ part} \quad (4)$$

The part of chaotic neuron is $\xi_i(t+1) + \zeta_i(t+1) + k_f\eta(t)$. And the part of synapse is $\sum_{j=0}^{n} w_{ij}x_j(t)$.

3.1 SC Chaotic Neuron Circuit

The SC chaotic neuron circuit with three-internal-state [7, 8] was proposed to realize the part of chaotic neuron. The circuit is shown in the Fig. 1.

In the chaotic neuron part of Eq. (2), k_r, k_f, k_e, α, θ_i and $\sum_{j=0}^{m} v_{ij}A_j(t)$ can be changed by capacitor value of C_{kr}, C_{kf}, C_{ke}, C_α, C_θ and C_e in switched-capacitor (SC) chaotic neuron circuits. The structure of these capacitor is used programmable capacitive array (PCA). PCA is technology that can change the value of capacitor by the digital signal. In this time, the digital signal is 8 bit in this system. Therefore, this chaos neural network system can freely change the value of the parameter in Eq. 3 within range of the specifications in the circuit.

3.2 Synapse Circuit

Then, we introduce a synaptic circuit that is used in the neural network system [9]. The i of Eq. 4 is changed like Eq. 5 to cope with 10,000 of neurons.
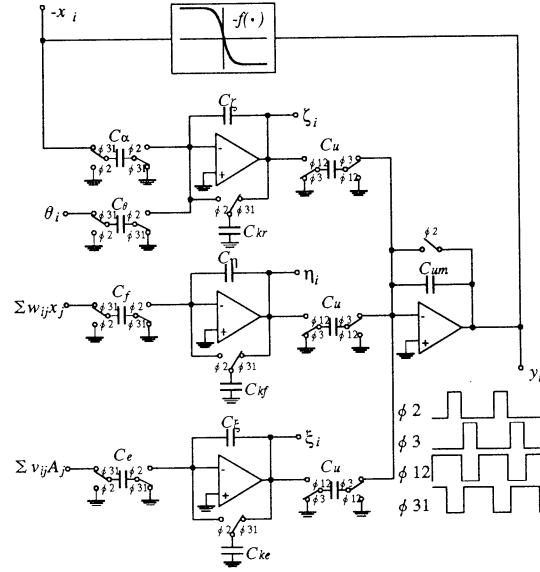
Figure 1: The analog neuron circuit

$$\sum_{j=0}^{9,999} w_{ij}x_j(t) \;=\; \sum_{a=0}^{99}\sum_{b=0}^{24}\underbrace{\sum_{c=0}^{3} w_{lj}x_j}_{Memory} \quad (5)$$

$$l \;=\; a \times 100 + b \times 4 + c$$

Where, n in the Eq. 4 is 9,999 because it is the total number of neurons in the system. And i separate a, b and c to reduce the number of the signals at the same time. In this case, we used 100 of signals at the same time in circuit. The diagram of the synapse which has function of Eq. 5 is shown in Fig. 2.

The synapse shown the diagram of Fig. 2 Are four modules such as memory, adder, accumulator and converter.

The converter is used to change to the specification of the neuron.

Each of memory modules have four weighted value is added, and it is into memory. So it corresponds with memory part in the Eq. 5. And using the memory module, the parameter of the part of synapse, w_{ij}, can be changed as well as the parameter of the part of neuron.

The adder module adds all of the value of memory module. And using accumulator module, signals at the same time is reduced.
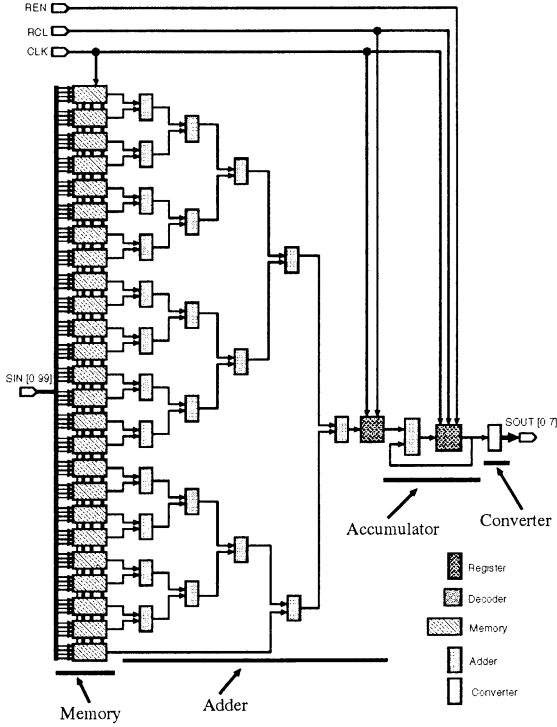
Figure 2: The digital synapse circuit

4 System Specifications

The specifications of the chaotic neural network system that introduce is described following

1. The clock speed of the neural network system is 10 [KHz].

2. The total number of neurons is 10,000 and each of the neurons is connected to all neurons.

3. The signal between each of neurons is binary such as 1 or 0.

4. The parameters of the system is a value which has a precision of 8 bits.

5. The Inputs data to neuron from synapse is 8 bits.

The first item mean that it can iterate Eq. 2 10,000 times for a second even if 10,000 of neurons is used. And, in this system, the clock of the synapse is over 1 [MHz] because the synapse circuits must operate quickly over 100 times.

In the second item, it means that the system has 100,000,000 synapse because each of the neurons must have a synapse every connected a neuron. In this system, number of neuron that has 10,000,000 of synapse is 10,000.

Because there is the third item, the synapse that is the interface between neurons are digital circuits.

The fourth items mean that all parameter in Eq. 2, k_e, k_f, k_c, α, θ, $\sum_{j=0}^{m} v_{ij}A_j(t)$ and w_{ij}, can change in the accuracy of 8 bits.

The final item mean that the process that calculated internal circuit of synapse is linear but when it is transfer to neuron, it is the data of 8 bits that the upper limit is limited.

5 System Structure

In Eq. 5, because the pins of the IC are limited and the propagation delay of the wire between IC become very large, the number of max value of inputs of the synapse defines as 100. If the number of inputs is the 10,000, it must input the 100 of inputs 100 times. Therefore, repetition of inputs correspond to the repetition of accumulator part, a, in the Eq. 5.

5.1 A Board Structure

So we propose the system which used 100 of boards that each of boards have 100 of neurons and 1000,000 of synapses. The number of the synapse must corresponds to 10,000 of neurons. If the only 100 of neurons are used, we need only the 10,000 of synapses. Fig. 2 is the board structure which is necessary for connection of the 100 board.
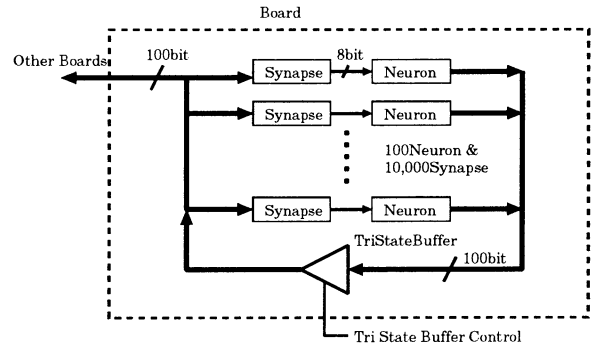
Figure 3: A Board Structure

The board has interface of 100 bit of inputs and outputs to connect other boards. The outputs of the external boards is directly connected to the inputs of synapse circuits. On the other hand, the outputs of the neuron circuits of the internal board is connected to inputs of synapse too, but it is not direct connection because signal through the tri-state buffer. The

output of tri-state buffer becomes high resistance or conductive condition by the control signal. The signal is controlled by external computer.

5.2 Communication Structure

Then we propose the method to communicate with all of the 100 boards which have 10,000 of neurons. The Fig. 4 is structured to transmit to the 100 bit of data from one board to other boards.
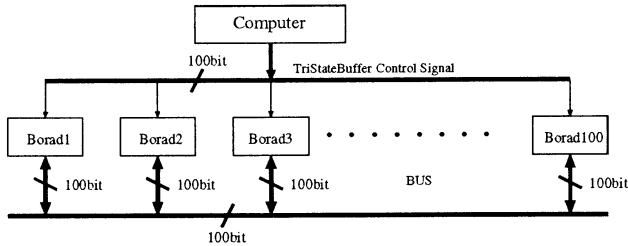
Figure 4: Communication structure for 100 of board

In Fig. 4, we assume that the output of the board of number 1 is enable and the outputs of other boards is disable by controlled Tri State Buffer Control signal in Fig. 3. The board of number 1 is called board1 for simply. The synapse of all boards can receive the 100 of neuron outputs of board1. Then board2 is enable and other board is disable. The synapses of all boards get the 100 of neuron outputs of board2. The 10,000 of inputs are realized by moving this operation to board100.

This bus condition which connected to all of boards and accumulator state in one synapse is shown by Fig. 4.
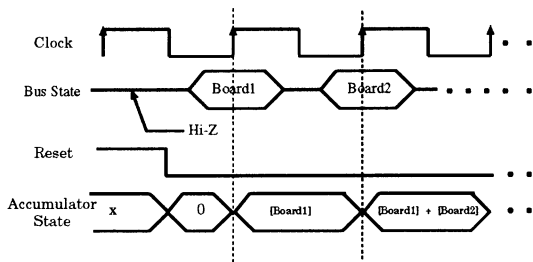
Figure 5: Communication data

6 Conclusions

We proposed the structure of chaotic neural network with switched-capacitor for 10,000 of neurons

and 100,000,000 of synapses in this paper. The SC chaotic neuron circuit with three-internal-state and the synapse circuit of 10,000 neurons are introduced. The network system which structure the neurons and the synapses can change the parameter that has accuracy of 8 bits in Eq. 2. So we expect that the system is applied to the various applications.

References

[1] K. Aihara, T. Takabe & M. Toyoda, "Chaotic Neural Networks," in *Phys. Lett. A*, vol. 144, pp. 333-340, 1990.

[2] M. Hasegawa, T. Ikeguchi and K. Aihara, "A novel chaotic search for combinatorial optimization," in *Proc. NOLTA '97*, pp. 613-616, 1997.

[3] M. Hasegawa, T. Ikeguchi and K. Aihara, "A novel approach for solving large scale traveling salesman problems," in *Proc. NOLTA '98*, pp. 711-714, 1998.

[4] M. Adachi and K. Aihara, "Associative dynamics in a chaotic neural network", *INNS Neural networks*, vol. 10, no. 1, pp. 83-98, 1997

[5] Y. Horio and K. Suyama, "Switched-capacitor chaotic neuron for chaotic neural networks," in *Proc. IEEE ISCAS'93*, pp. 1018-1021, 1993.

[6] Y. Horio and K. Suyama, "IC implementation of switched-capacitor chaotic neuron," in *Proc. IEEE ISCAS'94*, vol. 6 of 6, pp. 97-100, 1994.

[7] Y. Horio, I. Kobayashi, H. Hayashi and K. Aihara, "IC Implementation of a Multi-Internal-State Chaotic Neuron Model with Unipolar and Bipolar Output Functions" in *Proc. AROB'99*, vol. 1, pp. 90-93, 1999.

[8] H. Hayashi, H. Yoshihiko and K. Aihara, "Switched-Capacitor Chaotic Neuron Circuit with Three Internal States", in *Proc. NOLTA '98*, vol. 2, pp. 479-482, 1998.

[9] M. yoneda, O. Yamamot and H. Yoshihiko, "Synaptic Circuit for Large-Scale Chaotic Neural Networks", in *Proc. of the 1998 IEICE General Conference*, vol. 1, p. 28, 1998 (in Japanese).

Inter-Event Interval Reconstruction of Chaotic Dynamics

Isao Tokuda[†] and Kazuyuki Aihara[‡,§]

†: Dept. of Computer Science and Systems Engineering, Muroran Institute of Technology
‡: Dept. of Mathematical Engineering and Information Physics, Faculty of Engineering, University of Tokyo
§: CREST, Japan Science and Technology Corporation (JST)

Abstract

A network of stochastic resonance neurons is introduced to encode the topological structure of chaotic dynamics in "inter-event interval" of a coincidence detector neuron. We call the interspike interval of the coincidence detector neuron that detects *inter-synchronous-firing-event* of the network of stochastic resonance neurons "inter-event interval (IEI)." The numerical experiments demonstrate the network capability of encoding the chaotic dynamics efficiently in the IEI even in the presence of strong noise in the neural activity.

key words: Chaos, Neuronal coding, Interspike interval, Inter-event interval, Stochastic resonance

I. Introduction

Recent discovery and wide recognition of biological chaos [1, 2] have motivated our researches towards understanding possible functions of chaos in biological systems. In particular, problem for encoding the dynamical structure of chaos in the interspike intervals (ISIs) of a sensory neuron is quite interesting, since the ISIs have been recently focused on as an essential information carrier in the temporal neuronal coding [3, 4]. From this view point, ISIs of various neuron models activated by chaotic input have been recently intensively studied [5-7].

In case of a perfect integrator neuron that has no *leak* in the electrochemical potential, Sauer proved that the ISIs provide an embedding and that the topological dynamics of the original chaos can be exactly recovered in the ISIs [5]. Racicot *et al.* studied the ISIs generated from various neuron models such as the leaky integrate-and-fire neuron model and the FitzHugh-Nagumo neuron model and reported the difficulty of realizing the ISI reconstruction of chaotic dynamics by such realistic neuron models [6]. Castro *et al.*, on the other hand, observed significant improvement of the ISI reconstruction of chaotic dynamics by adding the stochastic resonance (SR) noise to excitable neuron models [7].

The present study focuses on the result of Castro *et al.* [7], since SR [10] has been recently focused on as a potential sensory mechanism of a neuron that detects weak input stimuli by small internal noise [11]. Our fundamental question to the result of Castro *et al.* is

How do the biological neurons tune their noise level?

In the stochastic resonance neuron, there is a small range of optimal noise levels that give rise to accurate sensory function. It might be hard for biological neurons to tune their internal noise level to such narrow noise range.

In order to widen the optimal noise range and to realize robust sensory mechanism, we introduce a network of SR neurons for ISI reconstruction of chaotic dynamics. We show that the network structure gives rise to robust sensory mechanism by suppressing the strong noise effect that destroys the geometric structure of the ISI reconstruction of chaotic dynamics.

II. FitzHugh-Nagumo neuron model

Let us consider the following FitzHugh-Nagumo (FHN) neuron model [8, 9] with SR noise:

$$\epsilon \dot{v} = -v(v - 0.5)(v - 1) - w + S(t) + \xi(t),$$
$$\dot{w} = v - w - e, \tag{1}$$

where v and w represent neuronal potentials, $\xi(t)$ represents *Gaussian* white noise with $E[\xi(t)] = 0$ and $E[\xi(t)\xi(s)] = 2D\delta(t - s)$, and e represents a system parameter whose value is fixed at $e = 0.15$. The neuron receives a weak *subthreshold* input: $S(t) = 0.075 + 0.0092x(t)$, from the $x(t)$-variable of the Rössler equations [12]: $\dot{x} = \tau(-y - z)$, $\dot{y} = \tau(x + ay)$, $\dot{z} = \tau[b + z(x - c)]$, where the parameters are fixed at $(a, b, c, \tau) = (0.36, 0.4, 4.5, 0.5)$.

In our numerical experiments, dynamics of the stochastic differential equation (1) is simulated by integrating the equation by the Euler's algorithm [13] with an integration step of $\Delta t = 0.001$.

In the FHN neuron model, the neuron fires and generates a single pulse when the neuronal potential v crosses a threshold value of $v_{th} = 0.7$. From the occurrence times of the neuronal firings $\{T_1, T_2, ..\}$, the sequence of ISIs can be obtained as

$$\{t_i = T_{i+1} - T_i : i = 1, 2, ..N_{\text{data}}\}. \tag{2}$$

By using the sequence of the ISIs, dynamical structure of the input chaotic stimuli is reconstructed in the d-dimensional delay-coordinate space [14]:

$$u(i) = (t_i, t_{i-1}, t_{i-2}, .., t_{i-d+1}). \tag{3}$$

In the present study, accuracy of the reconstruction of chaotic dynamics is quantitatively measured by the nonlinear prediction error (NPE) [16] as follows. First, we divide the ISI reconstruction dynamics, $\{u(i) : i = 1, 2, .., N_{\text{data}}\}$, into first and second halves. From the first half data, a nonlinear predictor $\tilde{f} : R^d \to R^d$ which approximates the data dynamics as $u(i+1) \approx \tilde{f}(u(i))$ is constructed. For the nonlinear predictor $\tilde{f}$, the Sugihara-May's local linear predictor [16] is exploited. Then, for the latter half data, nonlinear prediction is carried out. The forecasting procedure is that, for a give initial state $u(i)$, the s-step further state $u(i + s)$ is predicted as $\tilde{u}(i + s) = \tilde{f}^s(u(i))$ using the s-iterate of the predictor $\tilde{f}$. The NPE is finally computed as the following normalized root-mean-square error:

$$NPE = \frac{\sqrt{\sum_i \{u(i) - \tilde{u}(i)\}^2}}{\sqrt{\sum_i \{u(i) - E[u]\}^2}}. \tag{4}$$

In the numerical experiments, the reconstruction dimension, the prediction step, and the number of the data are set as $(d, s, N_{\text{data}}) = (4, 1, 5000)$.

Fig. 1 shows the NPE curve obtained from the FHN neuron with SR noise of $D \in [10^{-9}, 10^{-7}]$. We see that the response characteristics of the FHN neuron gives rise to a single sharp minimum which realizes an optimal ISI reconstruction of chaotic dynamics.

III. NETWORK OF FITZHUGH-NAGUMO NEURONS

Let us consider a network of FHN neurons with SR noise as follows (see also fig. 2):

$$\epsilon \dot{v}_i = -v_i(v_i - 0.5)(v_i - 1) - w_i + S(t)$$
$$+ \frac{C}{K} \sum_{j=1}^{K} (v_i - v_j) + \xi_i(t), \tag{5}$$
$$\dot{w}_i = v_i - w_i - e, \qquad (i = 1, .., K) \tag{6}$$
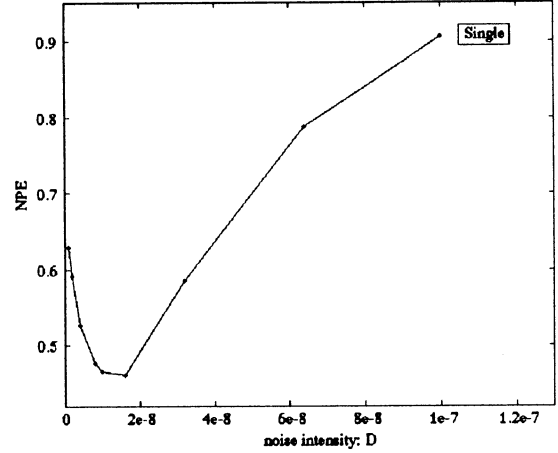
Figure 1: NPE of chaotic ISIs reconstructed by single FHN neuron with SR noise of $D \in [10^{-9}, 10^{-7}]$.
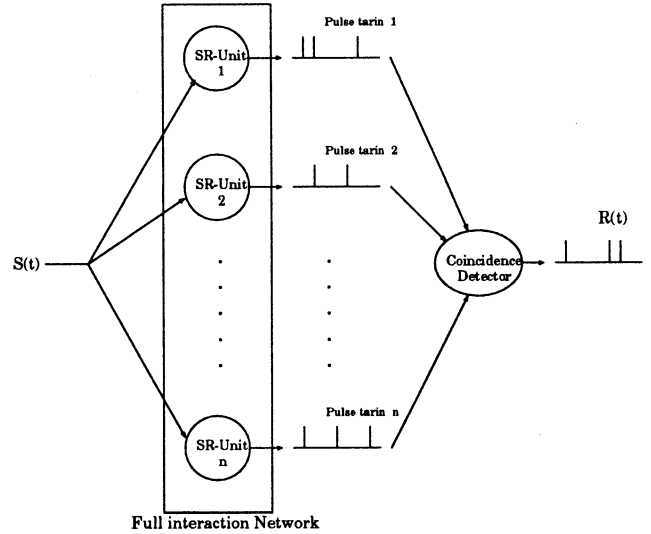
Figure 2: Schematic illustration of a network of FHN neurons with SR noise. The coincidence detector neuron receives pulse trains from all FHN neurons and fires when many pulse signals are received simultaneously.

where (v_i, w_i) represents a set of i-th neuron states, ξ_i represents i-th neuron's internal noise satisfying $E[\xi_i(t)] = 0$ and $E[\xi_i(t)\xi_j(s)] = 2D\delta(t - s)\delta(i - j)$, and K represents the number of the neurons. In order to realize synchronous firings between the FHN neurons, the coupling constant C is introduced to fully connect the neural elements.

In the network, every neuron v_i $(i = 1, .., K)$ receives a same chaotic input $S(t)$ and generates spike trains at the firing times $\{T_1(i), T_2(i), T_3(i), ..\}$. The coincidence detector neuron [15] receives the pulse trains from all of the neural elements and fires only when

the spike signals from more than κK neurons (κ: coincidence ratio) are simultaneously received within a coincidence duration of ρ. The detector also has an absolute refractory period of ν.

Since synchronous firing events of the FHN neurons give rise to firing times of the coincidence detector neuron, we call ISIs of the coincidence detector neuron "inter-event intervals (IEIs)." By using the IEIs $\{t'_i\}$, chaotic dynamics is reconstructed in the d-dimensional delay-coordinate space:

$$\boldsymbol{u}'(i) = (\ t'_i,\ t'_{i-1},\ t'_{i-2},\ ..,\ t'_{i-d+1}\ ). \qquad (7)$$
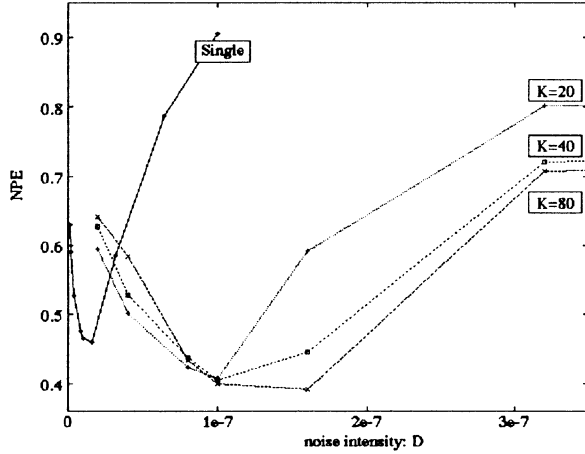
Figure 3: NPE of chaotic IEI reconstructed by a network of SR neurons with $D \in [10^{-9}, 10^{-6}]$. The number of the neurons is varied as $K = 20, 40, 80$.
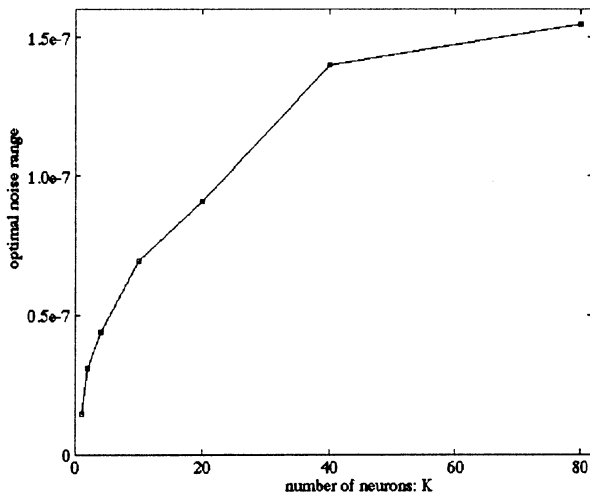
Figure 4: Optimal noise range is increased as the number of the neurons is increased from $K = 2$ to $K = 80$.

Fig. 3 shows the NPE obtained from the IEI-reconstruction (7) of a network of SR neurons with $D \in [10^{-9}, 10^{-6}]$. The parameters for the network and the coincidence detector are set to $C = 0.05$, $\kappa = 0.5$, $\rho = 0.25$, and $\nu = 0.5$. As the number of the neurons is increased from $K = 20$ to $K = 80$, range of the optimal noise levels that give rise to small $NPEs$ is widen significantly (see also fig. 4). This implies that the network structure gives rise to synchronous firings of the FHN neurons and suppresses the noisy component of the IEIs by selectively detecting the correct spike timings of the chaotic input.

IV. SUPRATHRESHOLD INPUT

Let us study the case of *suprathreshold* input: $S(t) = 0.1750 + 0.00920\ x(t)$. In Fig. 5, NPE curves obtained from the *suprathreshold* chaotic input are drawn. The solid line indicates the NPE curve of the ISIs of a single neuron model, whereas the dotted lines indicate the NPE curves of the IEIs of SR networks. For the ISI reconstruction by a single neuron model, the NPE monotonically increases as the noise level D is increased from 10^{-10} to 10^{-7}. This means that in case of *suprathreshold* input noise component included in the neuronal activity destroys the chaotic dynamical structure reconstructed in the ISI.
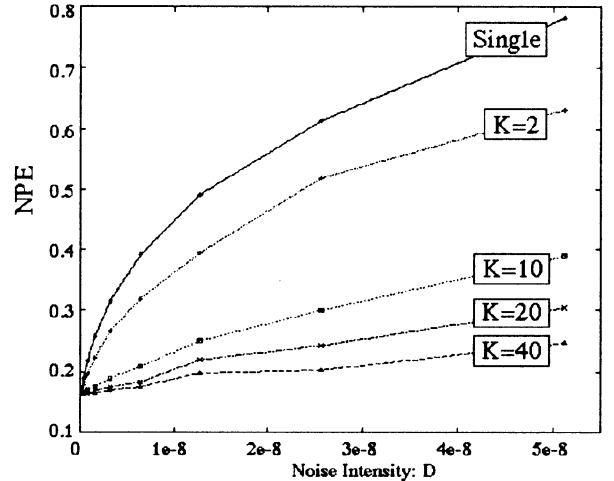
Figure 5: NPE of the IEI reconstruction of *suprathreshold* chaotic input with $D \in [10^{-10}, 10^{-7}]$. The solid line indicates the ISI reconstruction by single neuron model, whereas the dotted lines indicate the IEI reconstructions by networks of $K = 2, 10, 20, 40$ neurons.

As is shown by the NPE curves of the IEIs of SR networks, NPE is significantly decreased as the number of the SR neurons is increased. Fig. 6 shows 2-dimensional IEI-reconstruction of chaotic dynamics by a network of 1000 neurons with strong SR noise of $D = 5 * 10^{-8}$. Accurate reconstruction of the original

Rössler dynamics is discernible. This implies that the network structure works to suppress the noisy component of the IEI also for *suprathreshold* chaotic input.
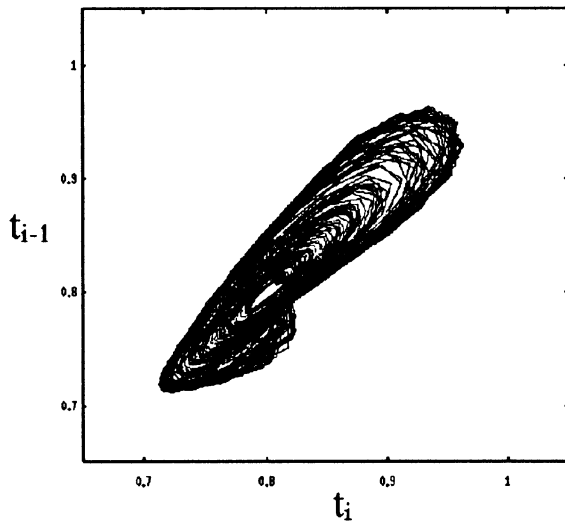
Figure 6: 2-dimensional IEI-reconstruction (t_i, t_{i-1}) of the Rössler dynamics obtained from a network of 1000 neurons with a noise intensity of $D = 5 * 10^{-8}$.

V. CONCLUSIONS AND DISCUSSIONS

A network of FHN neurons with SR noise is introduced to encode the topological structure of chaotic dynamics in the IEI of a coincidence detector neuron. The numerical experiments demonstrated the network capability of accurately reconstructing the chaotic dynamics in the IEI both from *subthreshold* input and from *suprathreshold* input. Coherent dynamics of the neural network efficiently works to suppress the incorrect spike signals induced by strong SR noise.

The significance of the stochastic resonance in sensory neuron is that a weak signal can be detected by small internal noise. We should be always careful, however, about the fact that the stochastic resonance neuron has to tune the noise intensity to rather small range of its optimal levels. It would be biologically more plausible if the optimal noise level exists in a wider range. For rate coding neurons, Collins *et al.* introduced a network of SR neurons for realizing a wide range of optimal noise levels [17]. The present study has shown that the idea of the network of SR neurons for realizing a robust sensory mechanism without careful tuning of the optimal noise level is also valid for the temporal coding neuron models.

In the present experiment, only a network of uniform neurons is considered. It is our further study to deal with non-uniform network composed of SR neurons with different neuron parameters and with different

noise levels. Physiological understanding of the possible function of the network of SR neurons activated by chaotic input is also an interesting and important open question [18].

REFERENCES

[1] H. Degn, A. V. Holden, and L. F. Olsen (Eds.), *Chaos in biological systems* (Plenum Press, New York, 1987).

[2] K. Aihara (Ed.), *Chaos in neural systems* (Tokyo Denki University Press, 1993).

[3] C. Von der Malsburg, *Synaptic plasticy as basis of brain organization: The neural and molecular basis of learning* (John Wiley, 1987).

[4] H. Fujii *et al.*, Neural Networks 9 (1996) 1303

[5] T. Sauer, Phys. Rev. Lett. 72(24) (1994) 3811

[6] D. M. Racicot & A. Longtin, Physica D 104 (1997) 184

[7] R. Castro & T. Sauer, Phys. Rev. Lett. 79(6) (1997) 1030

[8] R. FitzHugh, Biophys. J. 1 (1961) 445

[9] J. Nagumo, S. Arimoto, & S. Yoshizawa, Proc. IRE 50 (1962) 2061

[10] K. Wiesenfeld & F. Moss, Nature 373 (1995) 33

[11] A. Longtin, J. Stat. Phys. 70(1) 309

[12] O. E. Rössler, Ann. N.Y. Acad. Sci. 31 (1979) 376

[13] R. F. Fox *et al.*, Phys. Rev. A 38(11) (1988) 5938

[14] F. Takens, in *Lecture Notes in Math.* (Springer, Berlin, 1981), Vol. 898, p.366

[15] W. R. Softky, Neuroscience 58 (1994) 13

[16] G. Sugihara & R. M. May, Nature 344 (1990) 734

[17] J. J. Collins, C. C. Chow, T. T. Imhoff, Nature 376 (1995) 236;

[18] K. A. Richardson *et al.*, Phys. Rev. Lett. 80(11) (1998) 2485

On measuring distances between strange attractors

Tohru Ikeguchi
Department of Applied Electronics
Science University of Tokyo
2641 Yamazaki, Noda, Chiba 278–8510 Japan

Abstract

We propose an algorithm for measuring distances between strange attractors, or fractal sets. Considering cross–interpoint distances and estimating its frequency distribution, we propose a novel measure for evaluating the closeness or the difference of possible fractal measures. We show several examples of measuring distances between two finite sets with the novel method.

1 Introduction

Recent progress in experiments of nonlinear dynamics can easily provide us very long time series, such as 10^6 or more sampled points, which is followed by the embedding procedure that makes it possible to recover the underlying attractors through constructing the delay coordinates. As a result, many experimental researches have revealed possible existence of deterministic chaos during the last two decades, then we can understand complex behaviors in natural worlds from the view point of nonlinear low-dimensional dynamical systems and deterministic chaos.

For quantitative characterization of deterministic chaos, there are several statistics, for example, the fractal dimensions, the Lyapunov exponents, the metric entropies and so on. Among them, estimating fractal dimensions have still played an important role for evaluating characteristics of deterministic chaos, since fractal dimensions are one of the important measures for geometrical structure of (strange) attractors of dynamical systems. It is true that there are many dubious results obtained by applying fractal dimensional analyses in a blind way. However, if we use the measure very appropriately, we do believe that the fractal dimensions are still important for quantifying fractal structures of strange attractors of nonlinear dynamical systems.

There are several possible applications of fractal measures in the real world. For example, for analyzing complex time series with the theory of nonlinear dynamical systems, we often use nonlinear prediction methods for predicting the future behavior and evaluating characteristics of chaos, that is, short-term predictability and long-term unpredictability. For evaluating goodness of proposed prediction methods, we usually use the conventional measure, such as correlation coefficients or root mean square errors. In the case of evaluating one or two step ahead prediction performance, these measures are suitable for evaluating performance of nonlinear prediction methods. However, in the case of applying nonlinear prediction methods iteratively and producing points successively, which we call free-running, it is almost impossible to measure the goodness of prediction performance with correlation coefficients or prediction errors, since predicted points are actually different from true time series values due to sensitive dependence on initial conditions. Although there exists obvious difference with such numerical measures, we can see a kind of quantitative closeness between true attractors and freerun attractors, if the prediction method is appropriate for dynamics producing the time series. Then it is very important to introduce cross–measures for evaluating such kind of quantitative closeness.

Then, in this paper we propose an algorithm for measuring distances between strange attractors, or fractal sets by extending an estimating procedure of the fractal dimensions, or correlation dimension, based on calculating interpoint distances obtained from strange attractors to an estimation of cross–interpoint distances on two finite sets, which may be also a variant method to the previous one by Kantz [4].

2 A conventional measure through auto-interpoint distances

A basic algorithm for estimating dimension has already been proposed by Judd [3]. It is a maximum likelihood estimator, which calculates the parameter values of the distribution of interpoint distances.

First of all, the structure of attractors is assumed to be locally the cross-product of a self-similar (fractal) set and an absolutely continuous set [3]. This assumption may be true for many strange attractors in chaotic dynamical systems. For such structures, it is proven

that the distribution of interpoint distances calculated from a sample trajectory,

$$d_{ij} = |x_i - x_j|, \qquad (1)$$

has the following form of $P(\epsilon)$ for some $\epsilon_0 > 0$ which is referred to as the cutoff value:

$$P(\epsilon) \approx \epsilon^{\alpha+t}(a_0 + a_1\epsilon + \cdots + a_t\epsilon^t) \quad \text{for} \quad \epsilon < \epsilon_0, \quad (2)$$

where α and t are the correlation dimension of a self-similar set and the topological dimension of an absolutely continuous set, respectively. Therefore, the correlation dimension of the attractor D_C is the sum of α and t.

Firstly, interpoint distances on the points of the attractor are binned into small bins. Let the i th bin be denoted by $B_i = [\epsilon_{i+1}, \epsilon_i)$ for $i > 0$, where ϵ_i is defined as $\epsilon_i = \lambda^i \epsilon_0$, $\lambda < 1$ and the number of interpoint distances included in B_i are described by b_i. From Eq.(2), the number of interpoint distances included in B_i is calculated as,

$$
\begin{aligned}
p(b_i) &= P(\epsilon_i) - P(\epsilon_{i+1}) \qquad (3) \\
&= \epsilon_i^d(a_0 + a_1\epsilon_i + \cdots + a_t\epsilon_i^t) \\
&\quad - \epsilon_{i+1}^d(a_0 + a_1\epsilon_{i+1} + \cdots + a_t\epsilon_{i+1}^t),
\end{aligned}
$$

which is followed by

$$p_i = p(b_i) = \epsilon_i^d(q_0 + q_1\epsilon_i + \cdots + q_t\epsilon_i^t) \qquad (4)$$

where

$$q_i = (1 - \lambda^{d+i})a_i. \qquad (5)$$

Since b_i has a multinomial distribution given by p_i, then a maximum likelihood estimate for deciding α, t and q_i's is introduced by maximizing the log-likelihood of Eq.(6),

$$L(\alpha, t, q_i) = \sum_{i=1}^{\infty} b_i \log p_i, \qquad (6)$$

which has the constraints

$$\sum_{i=1}^{\infty} p_i = 1. \qquad (7)$$

3   A new measure through cross–interpoint distances

As stated above, the estimator by Judd works very well for fractal measures. Then, in this section, we extend the estimator to a new version by considering cross distance between two fractal sets.

Let us again rewrite the Judd's method in a different form. The Judd's estimator first estimate interpoint distance d_{ij} defined as follows:

$$d_{ij} = |x_i - x_j| \qquad (8)$$

where x_i is the i th sample from the set X. Then, to compare two sets X and Y, we extend the above auto-distance to the cross-distance defined below

$$D_{ij} = |x_i - y_j| \qquad (9)$$

where x_i is the i th sample from the set X, and y_j is the j th sample from the set Y.

If the points x_i and y_j come from the same sets, D_{ij}'s are perfectly equivalent to d_{ij}. If there is no relation between them, estimated values distribute differently, which lead to different estimated values. Then we calculate Eq.(9) instead of Eq.(8), we will obtain the relation between these two sets based on dimension estimation theory.

4   Numerical examples

4.1   Noisy strange attractors

In this paper, we utilize simple numerical examples for two finite sets. We take the Hénon map and the Ikeda map with standard parameter values. We add some amount of observational noise to obtained strange attractors, and evaluate the distance between them. In Fig.1, the set X is a sample trajectory of 2,048 discrete points of the Hénon map [1], and the set Y is its corruption with additive Gaussian random numbers of 10 [dB] in the signal to noise ratio. They are superimposed in Fig.1. Figure 2 is the same as Fig. 1, but for the Ikeda map [2]. In each figure, (a) is the phase portrait of the attractors and (b) is the estimated three interpoint distances; namely, auto-interpoint distances of the clean attractors and the noisy attractors, and cross-interpoint distance of both attractors.

In Figs.1 and 2, we can see that in the scaling range of $-3.5 < \epsilon < 0$, all of the interpoint distances exhibit almost the same properties. However, smaller than $\epsilon \approx -3.5$, the distribution of auto-interpoint distances for the clean attractor and the distributions of cross-interpoint distances and auto-interpoint distances for the noisy attractors show different scaling. The critical scale, in this case $\epsilon \approx -3.5$ can be defined as one of the closeness between these attractors.

4.2   Surrogate data of nonlinear dynamical systems

The method of surrogate data now becomes one of the essential scheme in the field of chaotic time series analysis. The basic idea of the method of surrogate data is that we can statistically extract the difference between the original time series and its surrogate time
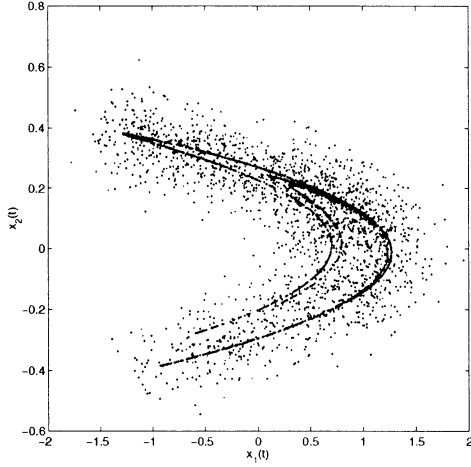
(a)
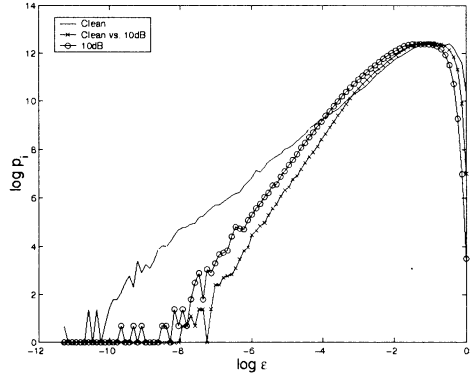
(b)

Figure 1: (a) An example of the Hénon map with 2,048 points, and the same data corrupted by 10[dB] additive observational noise. (b) The three interpoint distances from the data in (a).

(a)

(b)

Figure 2: (a) An example of the Ikeda map with 2,048 points, and the same data corrupted by 10[dB] additive observational noise. (b) The three interpoint distances from the data in (a).

series if it exists. The surrogate time series is one of the realization which preserves only linear structures but destroys nonlinear deterministic structures, so if we find any difference, we can conclude that the difference might come from the existence of deterministic structure which have been destroyed.

Usually the procedure in the method of surrogate data consists of the first part that makes the surrogate time series, and the second part that proceeds the comparison between the original time series and the surrogate time series. In this section, we apply the proposed scheme to the comparison.

In Fig.3, we show the comparison of the above case. We can clearly see that there are different scaling property between the original (the clean Ikeda map) and the surrogates. Since the surrogate data is almost distributed in two-dimensional state space (See Fig.3(a)),

the scaling property is more higher than the Ikeda map.

4.3 Freerun attractors

Nonlinear prediction is one of the important tools for analyzing complex time series data from the view points of nonlinear dynamical system theory. If we could construct a good model, we might have high nonlinear predictability even if the time series exhibits very complicated behavior. In this section, we will use the proposed strategy for evaluating the prediction performance of nonlinear prediction.

Figure 4 is the result of comparing the prediction performance of freerun prediction. They are the cases that a kind of local linear, (nonlinear) prediction scheme is utilized to predict the future behavior

(a)

(b)
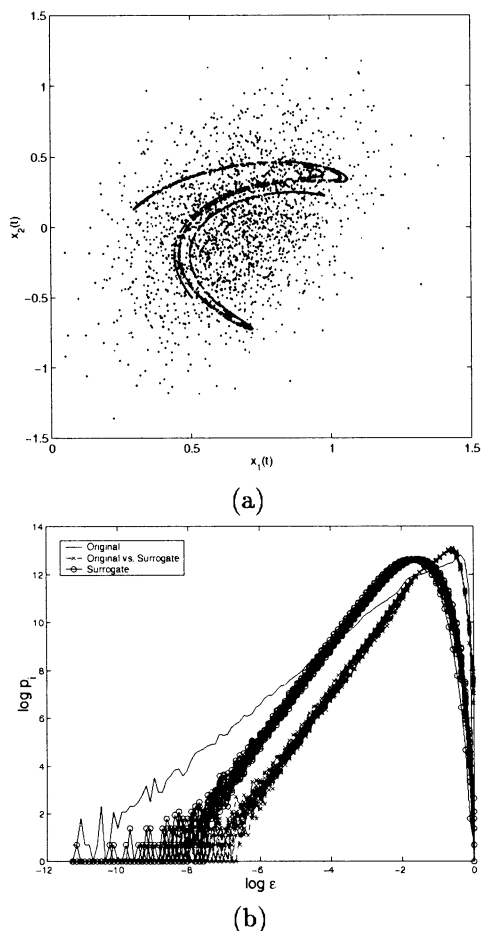
Figure 3: (a) An example of the Ikeda map with 2,048 points, and its two-dimensional FT surrogate. (b) The three interpoint distances from the data in (a).

(a)

(b)

Figure 4: Comparing nonlinear predictability of freerun prediction on the Ikeda map. Reconstructing attractors in (a)2 and (b) 4 dimensional state space.

of the Ikeda map with 1,024 points. In Fig.4(a) the reconstructed dimension is 2 and in Fig.4(b) is 4. It is acknowledged that for the Ikeda map, it is not possible to unfold the global structure of the attractors in two dimensional state space, then we can expect that the prediction performance in two dimension is not so good. From Fig.4(a), we can see that the scaling property of the freerun (crosses with the line) is almost the same in relatively larger scales, however, in the case of smaller than $\epsilon \approx -4.5$, they are different. From Fig.4(b), we can find that around $\epsilon \approx -6$ they show almost the same tendencies, which leads to the conclusion that for predicting the Ikeda map we should model it at least in four-dimensional state space.

5 Conclusion

We have introduced the cross–interpoint distributions to characterize the distance between two finite sets, and discussed the relation between them. We will extend the proposed scheme for evaluating the closeness of interspike interval time series for the future work. This research was partly supported by Grant-in-Aid from the Ministry of Education, Culture and Science of Japan (No.10750287).

References

[1] M. Hénon. *Comm. Math. Phys.*, Vol. 50, pp. 69–77, 1976.

[2] K. IKEDA. *Opt. Comm.*, Vol. 30, No. 2, pp. 257–261, 1979.

[3] K. Judd. *Physica D*, Vol. 71, pp. 421–429, 1994.

[4] H. Kantz. *Phys. Rev. E*, Vol. 49, No. 6, pp. 5091–5097, 1994.

Chaotic EEG Analysis and Judgment of Brain Death

GEN HORI[1], KAZU AIHARA[2], YOSHIKUNI MIZUNO[3] and YASUYUKI OKUMA[3]

[1] Laboratory for Open Information Systems, Brain Science Institute, RIKEN
2-1, Hirosawa, Wako-shi, Saitama 351-0198

[2] Department of Mathematical Engineering, University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656

[3] School of Medicine, Juntendo University
2-1-1, Hongo, Bunkyo-ku, Tokyo 113-8421

Abstract

Judgment of brain death based on flat EEG criteria has turned out to be difficult because signal from heart action and line noise contaminate EEG signals. Blind source separation is a good method for eliminating such contaminations from EEG signals. This paper proposes use of chaos indices of EEG signals, together with blind source separation method, as an aid for judgment of brain death.

Keywords

brain death, EEG(electroencephalogram), chaos, Lyapunov exponent, blind source separation

1 Introduction

Judgment of brain death for organ transplant is one of the most important social issues at present in Japan.

Judgment of brain death is based on clinical and EEG(electroencephalogram) criteria. EEG must be flat to conclude brain death. Japanese criteria of brain death is most strict in the world and "flat EEG examination" must be done in fivefold gain of EEG machine. It has been reported several times that, in fivefold gain, EEG of patients with clinically apparent brain death showed some weak signal presumed to be signal from heart action and line noise which made judgment of brain death very difficult.

Employing blind source separation method, signal from heart action (ECG, electrocardiogram) and line noise can be successfully eliminated from EEG. On the other hand, some chaos indices of EEG signals are known to reflect degrees of human brain activities and can be used as an aid for judgment of brain death.

2 Human brain activities and chaos indices of EEG signals

There has been a lot of works on chaotic time series analysis of EEG signals mainly using correlation dimension since Grassberger and Procassia[3] proposed a method for calculation of correlation dimension of real data by calculating correlation integral. Among them, many reported the relation between human brain activities and chaos indices of EEG signals

Rapp et al.[6] recorded EEG data of a subject who went through Rest→CalculationTask1→Rest→CalculationTask2→Rest and calculated the correlation dimension of the EEG data. The result was $3.4\pm0.7 \rightarrow 4.8\pm0.2 \rightarrow 4.1\pm1.0 \rightarrow 4.8\pm0.9 \rightarrow 4.3\pm0.4$ and showed positive correlation between the correlation dimension of EEG signals and human brain activities.

Dvořák et al.[1] compared EEG signals recorded on two points on the scalp O2 and C4 (see Fig.4.) of a subject i) with eyes closed, ii) with eyes opened and iii) under task. The correlation dimension of the EEG signals were calculated as i)5.7, ii)6.4, and iii)6.4 on O2, and i)5.5, ii)5.4, and iii)4.8 on C4. This result showed that the correlation dimension of the EEG signal recorded on O2 has positive correlation to human brain activities while the signal recorded on C4 has negative one.

Although many works were done on correlation dimension analysis of real EEG data, it has been pointed out that applying dimensional analysis for real data has some problems such as the decision of linear scaling region and the detection of spurious dimensions. On the other hand, the Lyapunov spectrum analysis of real data does not have these problems peculiar to dimensional analysis. Gallez and Babloyanz[2] calcu-

lated the Lyapunov exponent of real EEG data and Ikeguchi et al.[4] obtained all the Lyapunov spectrum of real EEG data.

3 Blind source separation method and noise elimination of EEG signals

Blind source separation is a method of multi-channel signal processing based on statistical independence of source signals. Let

$$s(t) = (s_1(t), \ldots, s_n(t))^t, x(t) = (x_1(t), \ldots, x_m(t))^t$$

be source signals and observed signals. The source signals $s_i(t)$ are all zero-mean and independent to each other. The observed signals are assumed to be linear mixtures of source signals

$$x(t) = As(t)$$

where mixing matrix A is an $m \times n$ real matrix. Blind source separation is a problem of finding demixing matrix W which recovers independent source signals

$$y(t) = Wx(t)$$

without using any knowledge on the mixing matrix A or the probability distributions of source signals. Demixed signals $y_i(t)$ are independent to each other and the matrix WA is a permutation matrix with amplitude.

Lee et al.[5] proposed a blind source separation learning rule

$$\dot{W} = (I - K\tanh(y)y^t - yy^t)W$$

where

$$
\begin{aligned}
K &= \text{diag}(k_1, \ldots, k_n) \\
k_i &= \text{sign}(E[\text{sech}^2(y_i)]E[y_i^2] - E[\tanh(y_i)y_i]).
\end{aligned}
$$

This algorithm utilizes auto-detection mechanism of sub-Gaussian and super-Gaussian sources.

Fig.1.-3. show an example procedure of noise elimination from EEG signals using the algorithm of Lee et al. The elimination of each noise source is done as follows: 1. find the demixing matrix W using the algorithm, 2. demix the channels to independent sources, $y = Wx$, 3. replace the source to be eliminated by all zeros, $y_k = 0$, and 4. remix the sources to channels, $x = W^{-1}y$. Fig.1. shows EEG signals of patient with presumed brain death in five fold gain, together with ECG(electrocardiogram, signal of heart action) channel. Fig.2. shows EEG signals with contamination from the ECG channel eliminated. Fig.3. shows EEG signals with contamination from line noise eliminated.

4 Calculation of Lyapunov exponent

Numerical calculation method of Lyapunov spectrum was introduced by Shimada and Nagashima[8], and extended for real experimental data by Sano and Sawada[7]. In this paper, we calculate the Lyapunov exponent of EEG data of the subject with presumed brain death following Ikeguchi et al.[4].

The calculation of the Lyapunov exponent of real experimental data is based on the estimation of the differential DF of the map

$$x_{t+1} = F(x_t)$$

which represent the dynamics of reconstructed orbit in the n-dimensional state space

$$x_t = (\xi(t), \xi(t + \tau), \ldots, \xi(t + (n - 1)\tau))$$

where $\xi(t)$ is a single channel observed data and τ is a delay time. Consider a set $\{x_{k(i)}\}(i = 1, 2, \ldots, M)$ consists of points in a small ball with center x_t and radius ε and displacement vectors at a interval of m time steps,

$$y_i = x_{k(i)} - x_t, \ z_i = x_{k(i)+m} - x_{t+m}.$$

The estimation A_t of the differential DF is a matrix which satisfies

$$z_i = A_t y_i \ (i = 1, 2, \ldots, M)$$

as far as possible and can be obtained through minimizing

$$S = \frac{1}{M} \sum_{i=1}^{M} |z_i - A_t y_i|^2.$$

The minimal condition

$$\frac{\partial S}{\partial A_t} = 0$$

yields

$$A_t = CV^{-1}$$

where (k, l)-th elements of matrices V and C are given using k-th and l-th elements of y_i and z_i as

$$v_{kl} = \frac{1}{M} \sum_{i=1}^{M} y_{ik}y_{il}, \ c_{kl} = \frac{1}{M} \sum_{i=1}^{M} z_{ik}y_{il}.$$

The maximum Lyapunov exponent of the observed data is calculated by multiplying the unit vector $u(t)$ by the estimated differential A_t,

$$e(t + 1) = A_t u(t), \ u(t + 1) = \frac{e(t + 1)}{|e(t + 1)|}$$

and averaging the growing rate of $u(t)$,

$$\lambda_{max} = \lim_{N \to \infty} \frac{1}{N\tau} \sum_{t=1}^{N} \log |e(t)|.$$

5 Numerical experiments

The EEG data of the subject with presumed brain death is recorded with a sampling rate 709.2Hz. The number of data points is 3000 and we use the central 2000 points for calculation. The maximum Lyapunov exponents of the EEG data is calculated with embedding dimension $n = 5$, delay time $\tau = 14$ and time step $m = 1$. The radius of small ball ε is set to 5% of the diameter of the embedded attractor and 20 nearest points to the center of the small ball is used for the estimation of the differential DF.

The results are given in Table 1. Our settings are very similar to Ikeguchi et al.[4] so that our results can be compared to the results given in Table 2.

Table 1. maximum Lyapunov exponents of EEG of subject with presumed brain death [sec^{-1}]

	Fig.1.	Fig.2.	Fig.3.
Fp1-A1	61.71	66.72	35.37
C3-A1	18.79	39.56	43.08
O1-A1	32.70	29.34	51.03
ECG	19.95	—	—

Table 2. Lyapunov spectrum of EEG of healthy subject [sec^{-1}] (Ikeguchi et al.[4])

λ_1	101
λ_2	-35
λ_3	-152
λ_4	-334
λ_5	-800

6 Concluding remarks

At first, all the maximum Lyapunov exponents of EEG signals of the subject with brain death in Table 1. are less than the maximum Lyapunov exponent λ_1 of the healthy subject in Table 2. This indicates the possibility that the maximum Lyapunov exponent and other chaos indices of EEG signals can be used as an aid for judgment of brain death.

The difference between EEG signals of Fig.1. and Fig.2. is contamination from ECG(signal from heart

action). This makes small difference of Lyapunov exponents on Fp1-A1 and O1-A1, but large difference on C3-A1.

EEG signals of Fig.3. are purest signals from brain activities among Fig.1.-3. Signals recorded on C3-A1 and O1-A1 show largest Lyapunov exponents in Fig.3.

Acknowledgments

References

[1] I.Dvořák and J.Siska, "On Some Problems Encountered in the Estimation of the Correlation Dimension of the EEG", *Phys.Lett.A*, 118, 2, pp.63-66, 1986.

[2] D.Gallez and A.Babloyanz, "Predictability of Human EEG : A Dynamical Approach", *Biol.Cybern.*,

[3] P.Grassberger and I.Procaccia, "Characterization of Strange Attractors", *Phys.Rev.Lett.*, 50, 5, pp.346-349, 1983.

[4] T.Ikeguchi, K.Aihara, S.Itoh and T.Utsunomiya, "An Analysis on the Lyapunov Spectrum of Electroencephalographic (EEG) Potentials", *Trans.IEICE*, E73, 6, pp.842-847, 1990.

[5] T.W.Lee, M.Girolami and T.J.Sejnowski, "Independent Component Analysis using an Extended Infomax Algorithm for Mixed Sub-Gaussian and Super-Gaussian Sources" (to appear on *Neural Computation*)

[6] P.E.Rapp, T.R.Bashore, J.M.Martinerie, A.M.Albano, I.D.Zimmerman and A.I.Mees, "Dynamics of Brain Electrical Activity", *Brain Topography*, 2, 1&2, pp.99-118, 1989.

[7] M.Sano and Y.Sawada, "Measurement of the Lyapunov Spectrum from a Chaotic Time Series", Phys.Rev.Lett, 55, 10, pp.1082-1085, 1985.

[8] I.Shimada and T.Nagashima, "A Numerical Approach to Ergodic Problem of Dissipative Dynamical Systems", *Prog.Theor.Phys.*, Vol.61, No.6, pp.1605-1616, 1979.

Fig.1. EEG of patient with presumed brain death
(fivefold gain)

Fp1-A1

Fp2-A2

C3-A1

C4-A2

P3-A1

P4-A2

O1-A1

O2-A2

F7-A1

F8-A2

T3-A1

T4-A2

ECG

Fig.2. ECG eliminated

Fp1-A1

Fp2-A2

C3-A1

C4-A2

P3-A1

P4-A2

O1-A1

O2-A2

F7-A1

F8-A2

T3-A1

T4-A2
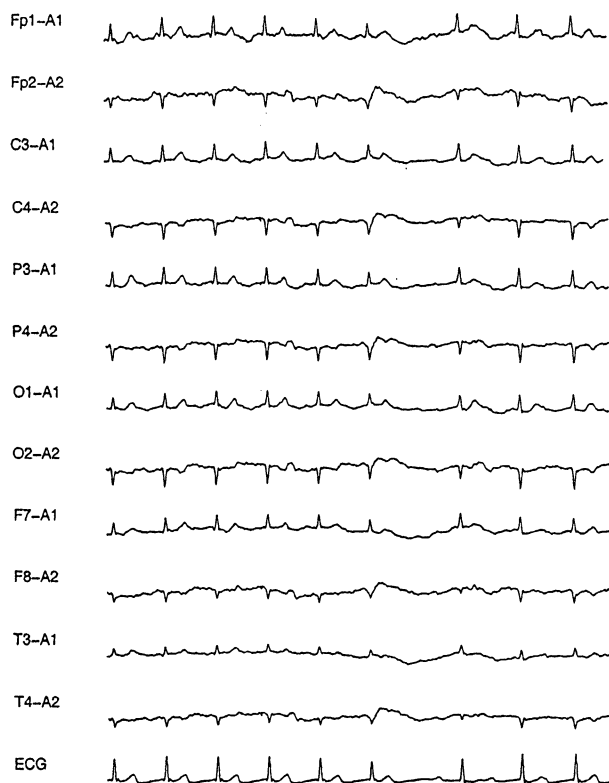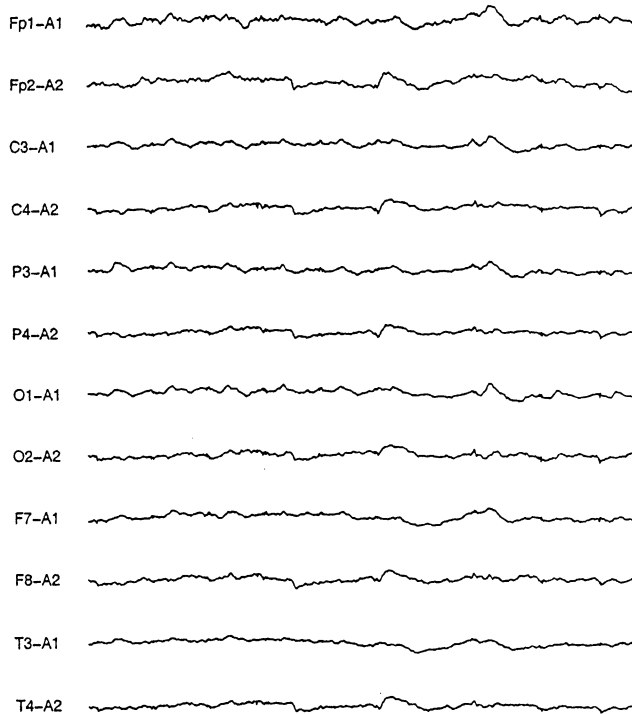
Fig.3. line noise on A1 and A2 eliminated

Fp1-A1

Fp2-A2

C3-A1

C4-A2

P3-A1

P4-A2

O1-A1

O2-A2

F7-A1

F8-A2

T3-A1

T4-A2

Fig.4.

FRONT
Nasion
Fp1   Fp2
F7   F3   Fz   F4   F8
A1   T3   C3   Cz   C4   T4   A2
LEFT   RIGHT
P3   Pz   P4
T5   T6
O1   O2
Inion
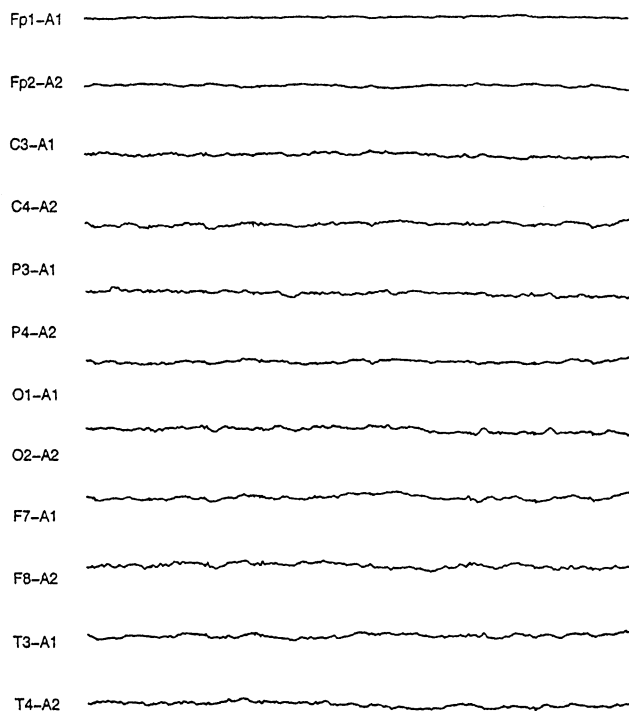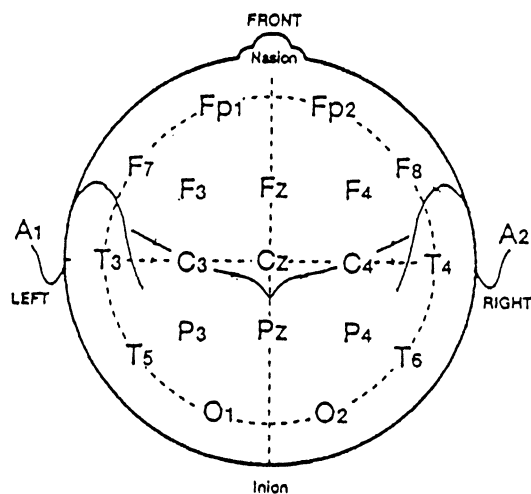
Exploring dynamics of interest rate by nonlinear time series approaches

Zhaoyun Shi   Yoshiyasu Tamura   Tohru Ozaki

The Institute of Statistical Mathematics
4-6-7 Minami-Azabu, Minato-ku, Tokyo 106-8569, Japan
Email: zyshi@ism.ac.jp

Abstract

The interest rate models in the literature have assumed that the drift is corresponding to a linear autoregressive process or constant. However, the argument for the question whether or not the drift is actually linear occurred in recent years. Regarding the fact that mean reversion of interest rate process is an important factor making models complicate, this paper introduces a new parameterized nonlinear short rate model, exponential drift model, which is potentially applicable to describing the mean reversion property of financial processes. Both of the new model and popular linear drift CKLS models are compared through empirical analysis of the Eurodollar short rate. The result shows the evidences of nonlinear drift in the short-term interest rates.

1. Introduction

The short interest rate plays a prominent role in many areas of finance. The modeling of the short rate represents a difficult and challenging problem. After an innovative work of Black and Scholes[1] on option pricing model, stochastic differential calculus has become a common tool to interpret price changes of financial instruments. It has been widely accepted that a short-term interest rate process can be described by the following continuous time stochastic differential equation.

$$dr_t = \mu(r_t, \theta)dt + \sigma(r_t, \theta)dB_t \qquad (1)$$

where $\{B_t, t \geq 0\}$ is a standard Brownian motion, the function $\mu(\cdot)$ and $\sigma^2(\cdot)$ are the drift and diffusion functions of the process, respectively, and $\{r_t, t \geq 0\}$ is interpreted as the level of the short-term interest rate. Equation (1) is applicable to characterizing the term structure of interest rates

and pricing derivative securities, however, it brings us a practical problem that is how to specify the drift and diffusion functions. It is of great importance to represent the drift and diffusion functions with appropriate parametric forms. Unfortunately, it seems an unanswered question to this task since there is no theory available for guidance. This paper gives a discussion on exploring appropriate models.

2. Short Rate Models

There are numbers of parametric model provided to specify the stochastic differential equation (1), in which a typical nesting one is the CKLS model[2] as follows.

$$dr_t = (\alpha_0 + \alpha_1 r_t)dt + \sigma r_t^\gamma dB_t \qquad (2)$$

We call equation (2) as the linear drift model. Model (2) gives an emphasis on the importance of correct assignment of γ to allow the conditional volatility of interest rate changes to be highly dependent on the level of the interest rate.

Recently, the identification of the short-term interest rate has greatly attracted the attention of both statisticians and financial theorists. From the viewpoint of avoiding possible misspecification of equation (1) by the parametric approach, nonparametric approach has been strongly proposed to estimate both the diffusion and drift functions. It is Aït-Sahalia[3], Stanton[4] and Jiang et al[5] who indicated that the linear drift assumption in model (2) is a main source of misspecification since they found evidence of high nonlinearity in the estimated drift function by applying the nonparametric kernel estimating technique to short rate data. In their investigation, the results both for daily 7-day Eurodollar deposit rate and Canadian daily 3-month treasury bill rates provide evidence for rejecting the common

parametric specifications for both drift and diffusion functions.

On the other hand, Pritsker[6] and Chapman & Pearson[7] argued that it is dangerous to conclude the nonlinearity of the short rate drift by applying the nonparametric estimators. They provided both theoretical and numerical simulation evidences to show that the nonparametric kernel estimators are easily misleading.

3. Dynamics of short-term interest rate

The dynamics of marketing process is complex and actually difficult to formulate. However, for short-term interest rate, an interesting phenomenon namely mean reversion seems very significant. Mean reversion is also seen in other typical marketing process like stock return, exchange rate, and so on. Mean reversion behavior makes marketing processes equilibrium, for example, whenever the interest rates exceed some range, they are pulled back into central region.

Most of the parametric models specify the drift function as a linear mean-reverting function, for example, in model (2), drift function $\alpha_0 + \alpha_1 r_t$ with $\alpha_1 < 0$ represents an attracting force that keeps pulling the rates towards its long-term mean α_0. The diffusion function in the model causes the process to fluctuate around the mean α_0 in an erratic but continuous fashion. The drawback of linear drift specification is that the mean reversion is inconsistent and thus fails to provide a desirable representation of the local-time behavior of the process. It is therefore desirable to develop a flexible parametric model available for describing reasonable dynamics of the short rate processes.

4. A new parametric model

Mean reversion can be actually regarded as a nonlinear behavior of a stochastic process. In central region, the interest rate process is more like a Brownian motion, whenever it escapes too far away from the equilibrium region, it should be strongly pulled back the middle region. So the mean reversion effectively guarantees the global stationarity of the process although it is locally nonstationary. In this sense, the mean reversion feature is very similar to the ergodic property for a stochastic process.

From this viewpoint, it seems reasonable to introduce some classic nonlinear time series models such as threshold and exponential autoregressive models in the specification problem

of equation (1). The advantage of those nonlinear model is amplitude-dependence which allows the models adaptively change with the state evolution.

In this study, we introduce the exponential autoregresive model to specify the drift function, thus we obtain a new parametric short rate model as follows,

$$\mu(r_t, \theta) = \alpha_0 + (\alpha_1 + \alpha_2 \exp(-\alpha_3 r_t^2))r_t$$
$$\sigma(r_t, \theta) = \sigma r_t^\gamma \quad (3)$$

where the diffusion function is still assumed as the same form of CKLS model (2). We call the model (3) as the exponential drift model. The unknown parameter vector is $\theta = (\alpha_0, \alpha_1, \alpha_2, \alpha_3, \sigma, \gamma)'$. We impose the following restrictions on the parameter values: $\gamma \geq 0$ and $\alpha_3 > 0$. Obviously seen that the exponential drift model will reduce to linear drift model whenever $\alpha_3 = 0$ and rewriting $\alpha_1 + \alpha_2 \to \alpha_1$. So the new model nests the linear drift model in a sense.

We estimate the parameters of both linear and exponential drift models by the maximum likelihood method. The performance of two models will be compared by AIC criterion.

5. Empirical analysis

The short-term rate used here is the 7-day Eurodollar deposit short rate, which has been used first by Aït-Sahalia[3]. The data set contains 5505 observations are daily from 1 June 1973 to 25 February 1995. A time series plot of the data is shown in Fig. 1. The explanation of the data refers to Aït-Sahalia[3]. The conclusion obtained by nonparametric estimators for this data that all the linear drift specification models are rejected.

We investigated the data by models (2) and (3). The goodness of fit for two models is checked by maximum log-likelihood $l(\hat{\theta})$ and AIC value, which are summarized as follows,

Model (2): $l(\hat{\theta}) = 24319$ AIC=-48630

Model (3): $l(\hat{\theta}) = 24322$ AIC=-48632

The above results mean that the exponential drift model (3) is more suitable than model (2) for describing the underlying data. We present the shapes of the drift and diffusion functions estimated from two models in Figs. 2 and 3. It is obviously seen in Fig. 2 that the exponential drift model surely extracted the mean reversion behavior

which is absent in linear drift model. Correspondingly, the difference between the diffusion functions of two estimated models is also visually significant due to the different drift function. This is important because the volatility evolution of the short rate is crucial in practical pricing contingent claims and hedging.
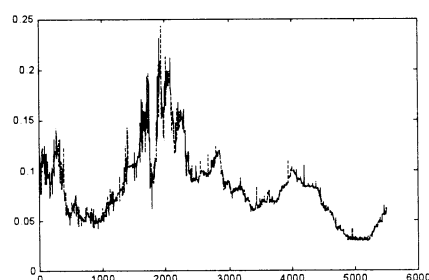
Fig.1 7-day Eurodollar rate
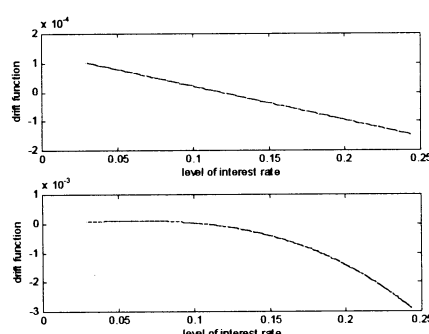
Fig. 2 Estimated drift functions of 7-day Eurodollar rate, upper is from model (2), below is from model (3)
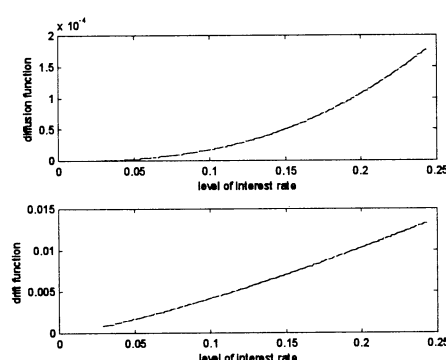
Fig. 2 Estimated diffusion functions of 7-day Eurodollar rate, upper is from model (2), below is from model (3)

6. Conclusion and discussion

We have introduced nonlinear time series models to analyze the short-term interest rate. The illustration supports our investigation to be meaningful. However, we have no intention to conclude that the drift function of short-term interest rate is nonlinear just like some nonparametric analysts indicated. We think that the conclusion might be different when one studies short rate of other contries or of different period. The evidences for the argument that precise claim of nonlinearity of the short rate is difficult to do have been found when we investigated other interest rate series.

On the other hand, we think from our experience, the results from nonparametric approach are usually unbelievable and unpracticable. To reach asymptotic property, nonparametric approach needs enough long samples, unfortunately, we often con not collect a long observations especially in finance, due to the short history of some financial derivatives. In this sense, to develop flexible parametric marketing model is a meaningful work.

References

[1] Black F, Scholes M (1973), The pricing of option and corporate liabilities. Journal of Political Economy, 81:637-654.

[2] Chan K C, Karolyi G A, Longstaff F A, Sanders A B (1992), An empirical comparison of alternative models of the short-term interest rate. Journal of Finance, 47: 1209-1227.

[3] Aït-Sahalia Y (1996), Testing continuous-time models of the spot interest rate. The Review of Financial Studies, 9: 385-426.

[4] Stanton R (1997), A nonparametric models of term structure dynamics and the market price of interest rate risk. Journal of Finance, 52: 1973-2002.

[5] Jiang G J, Knight J L (1997), A nonparametric approach to the estimation of diffusion processes, with an application to a short-term interest rate model. Econometric Theory, 615-645.

[6] Pristsker M (1998), Nonparametric density estimation and tests of continuous time interest rate models. Review of Financial Studies, 11: 449-487.

[7] Chapman D A, Pearson N D (1999), Is the short rate drift actually nonlinear? Journal of Finance, Forthcoming.

CoDi Technique: Cellular Automata as Neural Networks

Andrzej Buller

ATR Human Information Processing Research
Laboratories, 2-2 Hikaridai, Seika-cho,
Soraku-gun, Kyoto 619-02 Japan
abuller@hip.atr.co.jp
&
Technical University of Gdansk, Faculty of Electronics,
Telecommunications & Informatics,
ul. G.Narutowicza 11/12, 80-952 Gdansk, Poland
buller@pg.gda.pl

Katsunori Shimohara

NTT Communication Science Laboratories
2-4 Hikaridai, Soraku-gun, Kyoto 619-02 Japan
katsu@cslab.kecl.ntt.co.jp
&
ATR Human Information Processing Research
Laboratories, 2-2 Hikaridai, Seika-cho, Soraku-gun,
Kyoto 619-02 Japan
katsu@hip.atr.co.jp

Tomasz Chodakowski

Gdansk University,
Department of Mathematic & Physics,
ul. Wita Stwosza 57, 80-952 Gdansk-Oliwa, Poland
tch@manta.univ.gda.pl

Hitoshi Hemmi

NTT Communication Science Laboratories
2-4 Hikaridai, Soraku-gun, Kyoto 619-02 Japan
hemmi@cslab.kecl.ntt.co.jp

Abstract

We present an approach to building very large-scale neural systems, consisting of hundreds of millions of neurons, using cellular automata. The key concept is the CoDi module—a binary signal processing device to be evolved using a genetic algorithm in a cellular automata work space. We provide a formal definition of the CoDi module, results of tests of some CoDi modules evolved using a software simulator, as well as a basic description of the CBM—a special-purpose hardware for evolution of CoDi modules and running multi-module structures. We argue that the CoDi technique is a key to strong AI.

1 Introduction

The CoDi technique, developed in the framework of the ATR's CAM-Brain Project [2], combines neural engineering with cellular automata (CA) paradigm [11] and evolutionary computation [5]. The CoDi ("collect" and "distribute") model is based on CoDi cells—processing units defined in the framework of a cube of $24 \times 24 \times 24$ 3D CA cells. Every CoDi cell collects 1-bit signals obtained from some of its neighboring cells, processes them, and distributes among other neighboring cells. A set of CoDi cells is organized in such a way that they form the CoDi module—a model of a neural network receiving and producing spatio-temporal binary patterns [4]. Hence, the CoDi module belongs to the class of Pulsed Neural Networks [8].

A configuration of CoDi cells in a CoDi module is a function of a binary chromosome, and, this way, it may be a subject to an evolutionary process based on mutation and crossover.

Using a software simulator of CA and a genetic algorithm [3] several useful devices have been obtained, as for example, a timer triggered by a single pulse, frequency-to-delay converter, or an erasable-memory unit. Based on the last one, several kinds of learning systems can be built, including brain-like systems [1].

Using a special purpose hardware, called CBM, an evolution of a desired CoDi module should be possible in seconds, while a designed neural structure consisting of 64,640 CoDi modules (an equivalent of over 75 million neurons) should be updated almost 150 times per second. This means, that the CBM can serve an artificial brain controlling a behavior of a robot-pet, as, for example, a kitten [7].

2 CoDi Module

In order to define formally the CoDi module we will describe some auxiliary notions, as the CoDi cell, the CoDi neuron, and the CoDi pseudomodule. Let us also introduce the symbols: $\mathbf{B} = \{0, 1\}$, C – space of CoDi cells, $\mathbf{F} = \{-1, 0, 1\}$, $\mathbf{E} = \{i \mid i \in \mathbf{I}_+, i < 24\}$, $\mathbf{I}$ – space of integers, $\mathbf{L} = \{i \mid i \in \mathbf{I}, -8 < i < 2\}$, $\mathbf{I}_+$ – space of non-negative integers, $\mathbf{R}_+$ – space of non-negative real numbers, $\mathbf{T}$ – space of discrete values of flowing time (non-negative integers), and $\mathbf{V} = \{(v_x, v_y, v_z) \in \mathbf{F}^3 \mid v_x^2 + v_y^2 + v_z^2 = 1\}$.

CoDi cell is a basic processing unit receiving binary signals from and sending binary signals to its neighbors. It is convenient to considered it as a cube of the size $1 \times 1 \times 1$. Formally, it is a triple $\langle p, w, f \rangle$, where $p \in \mathbf{I}^3$ is the cell's position in a 3D space, $w \mid \mathbf{V} \to \mathbf{F}$ is a function returning values assigned to each of the six facets of the cell, $f \in \{\eta, \delta, \alpha\}$ is a *cell transition function* (to be described), and the following constraints are satisfied: (i) $f = \eta \Rightarrow (p \in \{ (x, y, z) \in \mathbf{I}^3 \mid x \bmod 3 = 0, y \bmod 2 = 0, z \bmod 2 = 0\}, \exists_{i, j \in \mathbf{V}}, i \neq j \quad w(i) = 0, w(j) = 0 \; \forall_k k \neq i, k \neq j, \quad w(k) \neq 0$), (ii) $f = \delta \Rightarrow \exists_{i \in \mathbf{V}}, w(i) = 0, \; \forall_k k \neq i \; w(k) = 1$, and (iii) $f = \alpha \Rightarrow \exists_{i \in \mathbf{V}}, w(i) = 1, \; \forall_k k \neq i \; w(k) = 0$.

If, for a CoDi cell, $f = \eta$, the cell is called *neural-body* cell. If $f = \delta$, the cell is called *dendritic* cell. If $f = \alpha$, the cell is called *axonic* cell.

Each of the CoDi cell collects signals through such of its facets that have assigned value -1 or 1 and sends signals through such of its facets that have assigned value 0. Some facets of a CoDi cell can be distinguished and referred as *gates* i.e. facets where a cell of a given type points to the object of its activity. A task of a *dendritic* cell is to collect signals from its five neighbors cells and distribute it to one neighbor cell its facet with assigned zero value points to. So in a *dendritic* cell the gate is such a facet i that $w(i) = 0$. A task of an *axonic* cell is to get signals from the neighbour its non-zero gate points to, and distribute them to its other neighbors. In an *axonic* cell a gate is such a facet i that $w(i) = 1$. In a *neural-body* cell can there are two and only two zero facets, which means that a neural-body cell can pass signals to exactly two neighbours. So in a neuron-body cell the gates the facets i and j such that $w(i) = 0$ and $w(j) = 0$.

Sender-receiver relation of two CoDi cells $c = \langle p, w, f \rangle$ and $c' = \langle p', w', f' \rangle$, denoted $\langle c \to c' \rangle$, where c is assumed to be a sender, while c' is assumed to be a receiver, takes place when $\exists_{i \in \mathbf{V}} \; p' = p + i, w(i) = 0, w'((0, 0, 0) - I) \neq 0$ and $(f, f') \in \{ (\delta, \delta), (\delta, \eta), (\eta, \alpha), (\alpha, \alpha), (\alpha, \delta) \}$. The relation has its characteristic function $\lambda \mid C^2 \to \mathbf{B}$, such that: $\lambda_{f \to f'}(c, c') = 1$ **iff** $c = \langle p, w, f \rangle, c' = \langle p', w', f' \rangle$ and $\langle c \to c' \rangle$, while otherwise $\lambda_{f \to f'}(c, c') = 0$, where C is the space of CoDi cells.

CoDi-cell stimulus is any function $u_c \mid \mathbf{V} \times \mathbf{T} \to \mathbf{B}$. The $u_c(i, t)$ describes a stimulus coming through a facet pointed by the vector i. A signal $u_c(t)$ is a reaction $y_{c'}(t)$ of the CoDi cell $c' = <p', w', f'>$ such that $p' = p + i$ and $\langle c' \to c \rangle$, i.e. $u_c(i, t) = y_{c'}(t)$. *CoDi-cell total stimulus* is any function $u_c \mid \mathbf{T} \to \mathbf{B}^6$ such that $u_c(t) = \{ (i, u_c(i, t)) \mid i \in \mathbf{V} \}$. At the time 0 all reactions and stimuli are set to zero i.e. $\forall_{c \in C, i \in \mathbf{V}} \; y_c(0) = 0, u_c(i, 0) = 0, u_c(0) = (0,0,0,0,0,0)$. *CoDi-cell state*, for a given CoDi cell c, is a function $x_c \mid \mathbf{T} \to \mathbf{L}$, such that if c is a dendritic or axonic cell, $x_c(t)$ is equal to the cell's reaction, while if c is a neuron-body cell, $x_c(t)$ is a kind of a counter

cumulating incoming signals. *CoDi-cell reaction* is a function $y_c \mid \mathbf{T} \to \mathbf{B}$ which depends on actual state of the cell c and on provided total stimulus. All possible CoDi–cell total stimuli, states and reactions constitute spaces denoted U, X and Y respectively.

A *CoDi-cell transition function* $f \mid U \to X \times Y$ is such that for a given CoDi-cell c, for all $u_c \in U$: $\forall_{t \in \mathbf{T}} \langle x_c, y_c \rangle = f(u_c), x_c(0) = 0, (f \in \{\delta, \alpha\} \Rightarrow y_c(t+1) = x_c(t+1) = \mathbf{OR}_{i \in \mathbf{V}}(w(i)u_c(i, t))), (f = \eta \Rightarrow y_c(t+1) = 1$ **iff** $net_c(t) \geq 2$ **while otherwise** $y_c(t+1) = 0$ **and** $x_c(t+1) = net_c(t)$ **iff** $net_c(t) \in \mathbf{L}$ **while otherwise** $x_c(t+1) = 0$), where $net_c(t) = x_c(t) + \Sigma_{i \in \mathbf{V}} w(i)u_c(i, t)$.

CoDi neuron n is a complex processing device consisting of CoDi cells. Formally, $n = \langle n, D, A \rangle$, where: $D = \{d_1, d_2, ..., d_n\}$ is a set of dendritic cells which form a dendritic tree of neuron body cell $n = \langle p, w, \eta \rangle \in C$ in such a way that: $\forall_{m=1,2,...,4} \; \lambda_{\delta \to \eta}(d_m, n)$ and $\forall_{k=..n} \exists_{k0, k1, ..., kj, m}, \; m \in \{1,2,...,4\} \mid \lambda_{\delta \to \delta}(d_k, d_{kj}) \wedge \lambda_{\delta \to \delta}(d_{kj}, d_{k-1}) \wedge ... \wedge \lambda_{\delta \to \delta}(d_{k0}, d_m)$. $A = \{a_1, a_2, ..., a_n\}$ is a set of axonic cells which form an axon of neuron body cell $n = \langle p, w, \eta \rangle \in C$ in a way such that: $\forall_{m=1..2} \; \lambda_{\eta \to \alpha}(n, a_m)$ and $\forall_{k=3..n} \exists_{k0, k1, ..., kj, m}, \; m \in \{1,2\} \mid \lambda_{\alpha \to \alpha}(a_m, a_{k0}) \wedge \lambda_{\alpha \to \alpha}(a_{k0}, a_{k1}) \wedge ... \wedge \lambda_{\alpha \to \alpha}(a_{kj}, a_k)$.

Two neurons n and n' are in neuron-level sender-receiver relationship, what is denoted by $\langle n, n' \rangle$, where the neuron $n = \langle n, D, A \rangle$ is called sender while $n' = \langle n', D', A' \rangle$ receiver, when $\exists_{a \in A} \exists_{d \in D'} \lambda_{\alpha \to \delta}(a, d)$.

CoDi pseudo-module m is a complex processing device consisting of CoDi neurons. Formally, $m = \{ n = \langle n, D, A \rangle \mid (\langle p, w, \eta \rangle = n, \langle p', w', \delta' \rangle \in D, \langle p'', w'', \alpha'' \rangle \in A) \Rightarrow p, p', p'' \in \mathbf{E}^3 \}$. *Active surface* of a CoDi pseudomodule m is a set A_m such that $A_m = \{p \mid p \in \{(0, y, z) \mid y, z \in \mathbf{E}\} \cup \{(x, 0, z) \mid x, z \in \mathbf{E}\} \cup \{(x, y, 0) \mid x_1, y_1 \in \mathbf{E}\}$.

Input-point set of a CoDi pseudomodule m is a set J_m of axonic cells such that: $J_m = \{ c = \langle p, w, \alpha \rangle \in C \mid p \in A_m, \; \forall_{v \in \mathbf{V}}, p+v \notin J_m \}$. *Output-point* set of a CoDi pseudomodule m is a set K_m of dendritic cells such that: $K_m = \{ c = \langle p, w, \delta \rangle \in C \mid p \notin A_m \}$.

CoDi module m is a complex processing device consisting of CoDi neurons configured in such a way that $m \in P, J_m \neq \emptyset, K_m \neq \emptyset$, where P is the set of CoDi pseudo-modules.

3 CoDi module characteristics

CoDi module is characterized by its quality measurement dependent on its transition function mapping a space of certain stimuli onto the space of its reactions.

CoDi-module local stimulus is, for a given CoDi module m, a function $s_m \mid J_m \times T \rightarrow B$, such that for all $i \in J_m$, $s_m(i, t) = u_i(t)$, where u_i is the total stimulus of the CoDi cell i. *CoDi-module total stimulus* is, for a given CoDi module m, a function $s_m \mid T \rightarrow B^J$, such that $s_m(t) = (s(i_1, t), s(i_2, t), \ldots s(i_J, t))$, where J is the number of input points. *CoDi-module state* is, for a given CoDi module m, a function $x_m \mid T \rightarrow I^C$, such that $x_m(t) = ((x_{c1}(t), x_{c2}(t), \ldots x_{cC}(t))$, where C is the number of all CoDi cells forming the CoDi-module neurons. *CoDi-module local reaction* is, for a given CoDi module m, a function $r_m \mid J_m \times T \rightarrow B$, such that for all $o \in K_m$, $r_m(o, t) = y_o(t)$, where y_o is the reaction of the CoDi cell o. *CoDi-module total reaction* is, for a given CoDi module m, a function $r_m \mid T \rightarrow B^K$, such that $r_m(t) = (r(o_1, t), r(o_2, t), \ldots r(o_K, t))$, where K is the number of output points. All possible CoDi-module total stimuli, states and total reactions form spaces denoted S, Z and R respectively.

CoDi module transition function is, for a given module $m \in M$, such a function $f_m \mid S \rightarrow Z \times R$, that $\exists f$ that for all $s_m \in S$: $\forall_{t \in T} \langle x_m, r_m \rangle = f_m(s_m) \Rightarrow \langle x_m(t+1), r_m(t+1) \rangle = f(x_m(t), s_m(t))$.

A *CoDi module* is to process a set of binary stimuli incoming in discrete moments over a period of time towards desired reactions. A quality of the processing, assumed to be equal to a quality of the CoDi module itself, may be measured based on observed reactions of the module to arbitrary designed stimuli.

CoDi module quality measurement is, for a given module m, spike-train length Δ, 'technological' delay τ, and an arbitrary designed set $\Phi = (\Phi_1, \Phi_2, \ldots, \Phi_n)$ of testing series of total stimuli, given for $t = 1, 2, \ldots, \Delta$, the value of the expression
$$\Sigma^n_{i=1} (1 - (\Sigma^K_{k=1} H(\omega_{m,k,i}, \theta_{m,k,i}, \Delta) / \Delta) / K) / n$$
where K is a number of used output points, $\theta_{m,k,i}$ is a desired spike-train to be produced at the k-th output point of m as a reaction to Φ_i measured for $t = \tau + 1$, $\tau + 2, \ldots, \tau + \Delta$, $\omega_{m,k,i}$ is a spike-train obtained at the k-th output point of m as a reaction to Φ_i, measured for $t = \tau + 1, \tau + 2, \ldots, \tau + \Delta$, and H is the Hamming distance between the two spike-trains.

4 What can CoDi module do?

A number of CoDi modules have been obtained using a software simulator of cellular automata. In order to obtain a CoDi module behaving in a desired way a genetic algorithm was implemented. And so, a population of random chromosomes was generated, based on the chromosomes CoDi modules were grown and tested, the most promising solutions could mate and have their offspring inheriting appropriate parts of their

parental chromosomes. Usually after passing of some hundreds of generations a satisfactory module appeared. This way we evolved, among others, a neural frequency-to-delay converter and a neural erasable memory unit.

The obtained converter works according to the principle 'the lower frequency of incoming spikes, the longer time of waiting for start of spike production'. Its quality measurement was taken based only on training examples and its value was equal to 0.974.

The erasable memory unit was intended to start producing a dense spiketrain when a not interrupted series of 4 spikes are provided to one of its inputs, and to stop producing spikes when a not interrupted series of 4 spikes are provided to the second of its inputs. In fact, each of the input spiketrains was provided to 6 arbitrary selected input points of the active surface of the CoDi module. For the erasable memory unit a quality measurement was taken based on a set of 32 testing examples not used during evolution and its value was equal to 1.0. Further investigation revealed an interesting property of the evolved memory unit. It proved to react properly to not interrupted series of three spikes or even to two adjacent spikes.

The SIIC (Spike Interval Information Coding) [9] and HAS (Hough Spiker Algorithm for Deconvolution) [6] can be used to convert analog waveforms to spiketrains and vice versa. This way CoDi can be used to a processing of 2D moving pictures [3].

5 CBM – a tool for CoDi engineering

The CBM (Cellular [Automata-Based] Brain Machine), developed in Genobyte Inc. (Boulder, Colorado), is a research tool for rapid CoDi-module evolving, as well as for the simulation of multi-module systems. A neural system, supported by the CBM, may consist of up to 64,640 CoDi modules, each populated by up to 1,152 neurons, for a total of 74,465,280 neurons. The CBM consists of the following six major blocks: CA, Genotype/Phenotype Memory, Fitness Evaluation Unit, Genetic Algorithm Unit, Module Interconnection Memory, and External Interface.

Genotype/Phenotype Memory is organized is such a way that each of the 72 FPGA daughter boards includes 16 Mbytes of EDO DRAM. A chromosome for a CoDi module, used in evolution phase, is a 90,001-bit word. In the run phase, the memory stores information concerning the locations and orientations of the neurons inside the CoDi module, and their synaptic mask.

Every module can produce simultaneously up to four different 96-bit long spiketrains to be taken from four different output points, stored in the *Module Interconnection Memory*, and then provided to all of the supported 64,640 CoDi modules. Every module can receive spiketrains from 188 other modules.

Cellular Automata (CA) is the hardware core of the CBM. It is intended to accelerate the speed of a large-scale neural net evolution through a highly

parallel execution of cellular state updates. It is implemented with new Xilinx FPGA devices XC6264BG560. Each of them contains 16,384 logic function cells, each cell featuring a flip-flop and Boolean logic capacity, capable of toggling at a 220 MHz rate. The CBM's CA consists of an array of identical hardware logic circuits arranged as a 3D structure of 24 * 24 * 24 cells—a total of 13,824 cells, all updated simultaneously with the speed 131 billion cells per second, which means that the state of a system consisting of almost 75 million neurons changes about 150 times per second.

6 Concluding Remarks

Combining neural engineering with CA and evolutionary computation gave us the CoDi technique—an extraordinary way of obtaining very large neural systems, consisting potentially of hundreds of millions of neurons. The systems are based on CoDi modules bred in a workspace of Cellular Automata (CA) using a genetic algorithm.

We have presented a formal description of the CoDi module and the behavior of some modules we evolved using a software simulator of CA. The results look quite promising, especially important is the evidence, that the CoDi module can serve as an erasable memory unit.

We have described the CBM—a special purpose hardware for rapid evolution of CoDi modules and running very large-scale multi-module systems. Because of the number of simulated neurons and the speed of the simulation, the systems can control autonomous robots or imitations of animals in near real-time regime. The potential scalability of CoDi-based systems makes them a key to human-like robots, or, one day, maybe super-humans.

It may be noted, that some applications of hand-designed AI systems are more impressive than CoDi modules we have evolved so far. Nevertheless, despite the learning capacity of the systems, they have their limits. We say, neither the wonderful recognizers of hand-written script, nor hand-designed networks for artificial drivers, but rather the CoDi technique, implemented using evolvable hardware, will open the gate to an artificial thinking entity. The reason is two-fold. First, the CoDi-based systems are enormously scalable. The possibility of adding new sets of modules to working CoDi-based systems is practically unlimited. Second, let us note, that CoDi modules have their binary chromosomes, hence, any CoDi-based autonomous agent has its binary 'genome'. This means that a population of CoDi-based agents can evolve over generations. In any case, the CoDi engineering has a good chance to become a revolution in computer engineering, while the CoDi modules have a good chance to become as ubiquitous as the microprocessor today.

References

[1] Buller A & de Garis H (1998) Brain Building Strategy: Some Remarks and Questions, *Intelligent Information Systems VII, Proceedings of the Workshop, Malbork, Poland, June 15-19*, 188-193.

[2] de Garis H (1995) "CAM-Brain": ATR's Billion Neuron Artificial Brain Project. A Three Year Progress Report, in: Furuhashi T & Uchikawa Y (Eds.) *Fuzzy Logic, Neural networks, and Evolutionary Computation*, Springer: Berlin, 215-243.

[3] de Garis, Nawa NE, Buller A, Korkin M, Gers.F & Hough (1999) Evolution of Neural Network Modules: ATR's Artificial Brain Project, in: Jain LC (Ed.) *Evolution of Engineering and Information Systems and Their Applications*, CRC Press: Boca Raton, 231-274.

[4] Gers F, de Garis H & Korkin M (1997) CoDi-1Bit: A Simplified Cellular Automata Based Neuron Model, *Evolution Artificiele 97, 22 Octobre 1997, Nimes, France*, 211-229.

[5] Goldberg DE (1989) *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley: New York.

[6] Hough M, de Garis H, Korkin M, Gers F & Nawa NE (1999) SPIKER: Analog Waveform to Digital Spiketrain Conversion in ATR's Artificial Brain (CAM-Brain) Project, Proc. Of the 4th Intl. Symposium on Artificial Life and Robotics (AROB'99), January 19-22, Oita, Japan, 610-613.

[7] Korkin M, de Garis H, Nawa NE & Rieken WD (1999) ATR's Artificial Brain Project: CAM-Brain Machine (CBM) and Robot Kitten (Robokoneko) Issues, in: Dobnikar A, Steele NC, Pearson DW & Albrecht R (Eds.) *Artificial Neural Nets and Genetic Algorithms*, Springer: Wien, 107-110.

[8] Maass W & Bishop CM (Eds.)(1999) *Pulsed Neural Networks*, A Bradford Book/The MIT Press: Cambridge, Mass.

[9] Rieke F, Warland D, de Ruyter van Steveninck & Bialek W (1997) *Spikes: Exploring the Neural Code*, A Bradford Book/The MIT Press: Cambridge, Mass.

[10] Shimohara K & Hemmi H (1999) [Evolving Artificial Brain] (in Japanese) *Computer Today*, 4, 4-9.

[11] Toffoli T & Margolus N (1987) *Cellular Automata Machines. A new environment for modeling*, The MIT Press: Cambridge,Mass.

An Interactive Mechanism for CAM-Brain

Hitoshi Hemmi, Tomofumi Hikage and Katsunori Shimohara
Evolutionary systems group
NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0237 Japan
{hemmi, hikage, katsu }@cslab.kecl.ntt.co.jp

Abstract

This paper proposes a new utilization method of network structure that enables a high-speed artificial brain system to evolve or learn in a relatively slow environment. The proposed structure is based on the feedback system in neurons, so the internal changes in the artificial brain caused by input signal trains to it remain for some duration. The structure is so natural that its operations may be considered similar to those of living creatures. In the more advanced form of this mechanism, the artificial brain contains an internal automaton that mimics part of the external environment, and during that time the artificial brain can be considered to be "dreaming."

1 Introduction

Neural network based control mechanisms are considered very flexible and robust. To make such a network dynamically configurable and genetically evolving, we have been investigating a cellular automata (CA) based neural network system called CAM (Cellular Automata Machine)-brain ([1], [2], [3], [4], [5]). In the CAM-brain system, the axons and dendrites of the network are implemented as genetically specified trajectory-like patterns in the CA space, and the signal transitions in them are realized as propagation of the CA states along with the pattern. The recently developed "CBM" (CAM-Brain Machine) is a direct and high-speed implementation of CAM-brain in electronics to provide a real-time application ([4]).

For the learning([5]) or evolution of the CBM, two different methods can be considered: off-line learning/evolution and on-line learning/evolution. In off-line learning/evolution, a limited number of sample input signals are used for training/trial. After it completes the adaptation, the CBM is employed in an actual control operation with applied practical inputs. In this circumstance, the CBM's electronic speed is expected to be useful for fast learning/evolution.

In on-line learning/evolution, such adaptation is done by using real inputs. However, there is a fundamental problem in this approach: in this process the CBM may not be able to make full use of its operating speed. For example, when the CBM is used for robot control, the mechanical systems such as motor control of leg/arm movement and control of the vision camera system may be very slow compared to CBM's electronic operation speed. Consequently, most of the time CBM will be simply waiting for the input signals to come in. In addition, because it uses the learning or evolution scheme, the CBM needs a lot of training/trials, maybe hundreds or thousands or even more, to adapt to the environment. It takes a long time. Therefore, in the final analysis CBM might not be efficient or suitable for a real-time on-line adaptive system.

The key to solving this difficulty lies in the utilization of the CBM's "leisure time," i.e., the time period when no important signal comes in. That is, during such leisure time the CBM should learn or evolve to adapt input signal trains that have already come in. A caveat is that this adaptation should be done without using explicit signal recording devices. If we use such devices, the CBM needs to switch its inputs from the external environment to the recording devices for adaptation, and it would no longer be an on-line adaptation; in other words, CBM could not deal with sudden input stimulus. Instead, it is better for the CBM itself to store its inputs internally in some form to allow its attention to autonomously wander among the current inputs and "afterimages" of previous inputs. This paper discusses network structures that make such operation possible.

It is worthwhile to note that a similar process may occur in living creatures. For example, we humans sometimes have to adapt to a problem that occurs very infrequently. A typical way to cope with such a situation is to do "imaginary training," that is, training

without an actual stimulus.

The next section briefly describes the CAM-brain. A discussion of the network structures follows.

2 CAM-Brain

CAM-brain is a three dimentional CA based neural network system. In the CA space of CAM-brain, there are several "neuron cells". In "growing phase" of CAM-brain, according to genetic informations distributed in the cells in the CA space, some trajectory-like cell patterns corresponding to axons and dendrites grow from each neuron cells. When an axon meets trajectory of a dendrite, the growth of the axon stops and a synapse is formed. Same thing happens when opposite case occurs.

In "signaling phase," the signal (pulse) moves one cell per a clock to the specified direction in the trajectory. When a signal comes along the axon to the synapse, it is transformed to the dendrite. Each neuron cell has a counter which count up when an excitatory signal comes in from a dendrite and count down if the signal come in is inhibitatory. When the counter reaches a preset threshold values, the neuron "fires" and send one pulse signal to each axon connected to the neuron. Simultaneously the counter reset to zero.

Figure 1 shows an example neural network in the CAM-brain.
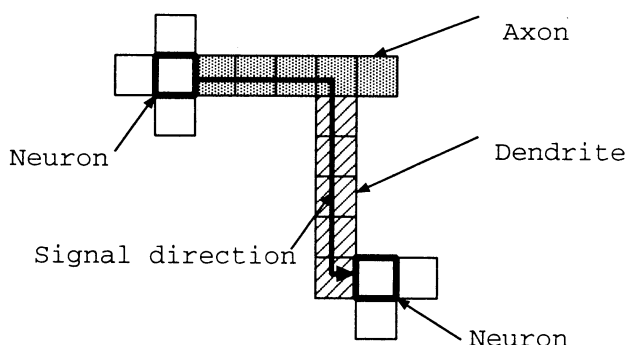
Figure 1: Trajectory in CAM-brain

3 Network structures

3.1 General Aspect of Networks

This subsection describes a graph theoretical view of neural networks. The facts described here are general and may be commonly known. Indeed, it should be emphasized that the network structures shown in the following subsections are not elaborate ones but easily constructed by natural/artificial evolutionary processes.

A network consists of nodes (neurons) and edges (axons and dendrites) connecting them. Here, consider so called "strongly connected components"; within a component (the set of neurons), any two neurons, directly or indirectly, give and take signals between each other (i.e., there is a signal feedback loop going through the two neurons). On the other hand, by regarding each component as one chunk, the network can be viewed as a network of the components. From this perspective, there is no longer a feedback loop among the components; every signal transfer is one way between any two components. Accordingly, the network can be viewed as signal "flows" among the components (Fig. 2).
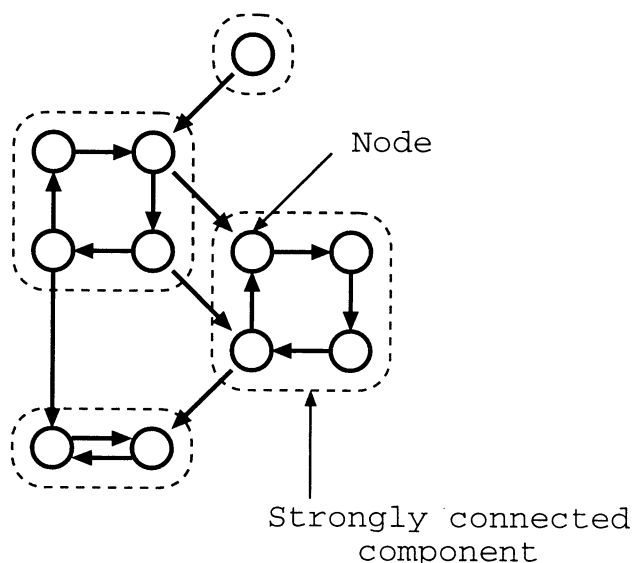
Figure 2: Strongly connected components and flows among them

In each component, find the longest feedback loop. Then the component can be viewed as having a structure composed of a big signal transfer loop of neurons plus some shortcut paths connecting the neurons in the big loop (Fig. 3). From this view point, we need to pay attention mainly the relationship or order of plural signal arrivals.

If we look at only part of the network, some feedback loops are not explicitly apparent; they look as if they were flows of several small components, each of that, in fact, is not an independent component (Fig. 4). This local view is sometimes useful. Be-
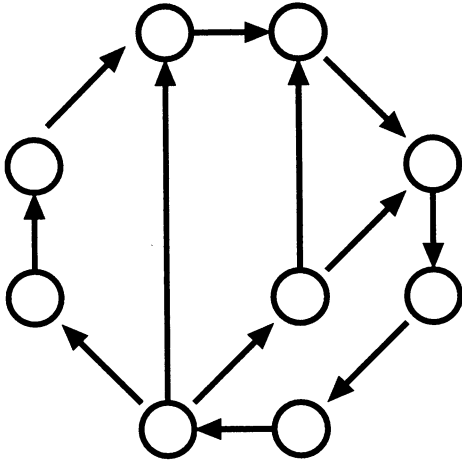
Figure 3: Structure of a component

cause there are some cases in which few signal travel round the big loop in actual; almost all signals may be decaied or cancelled.
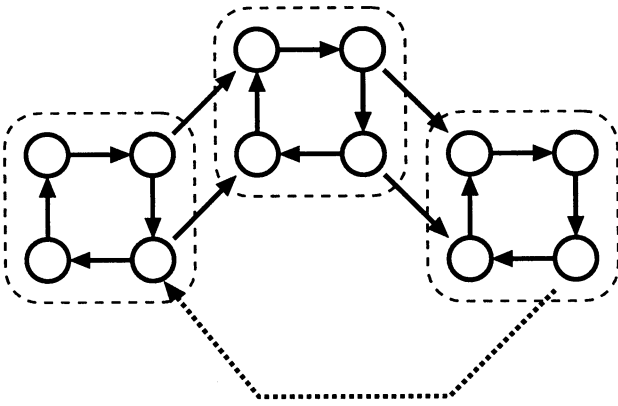
Figure 4: *Pseudo* independent components

3.2 Feedback loops that hold signal trains

This subsection shows example loops that hold signal trains. As previous subsection discussed, a large majority of the networks are naturally expected to contain loops (strongly connected components), and the operation of the loops could be understood from the signal delay.

First, consider the loop in Fig. 5. The signal transfer functionality is very simple even if it consists of some neurons. Assuming that no signal exists in the loop, this loop can repeat any signal train that is shorter than the length of the loop's circumference.

Apparently, if some signals already exist in the loop, it repeats overlapping of old and new signal trains.
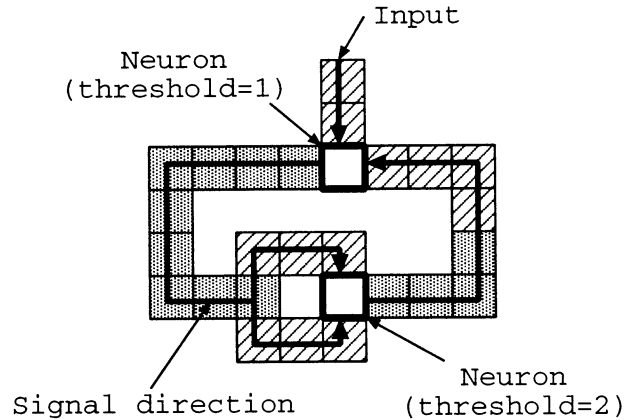
Figure 5: Simple loop

Next, consider the loop in Fig. 6. This loop selectively holds a two-pulse signal train pattern. The operation of this structure is as follows. If the lag of two pulses in the input signal train just corresponds to the difference of the length of two signal paths, the neuron will get concentrated stimulus and fires. Then each axon send out a pulse from the neuron. After a while these pulses will come back to the neuron, and same thing will happen. As the neuron in CAM-brain uses signal count up to a threshold scheme for firing, slight discrepancy in the input signal train will be allowed. However, if the signal train does not similar at all to the ideal one, signals are not concentrated and the counter decay ([5]).
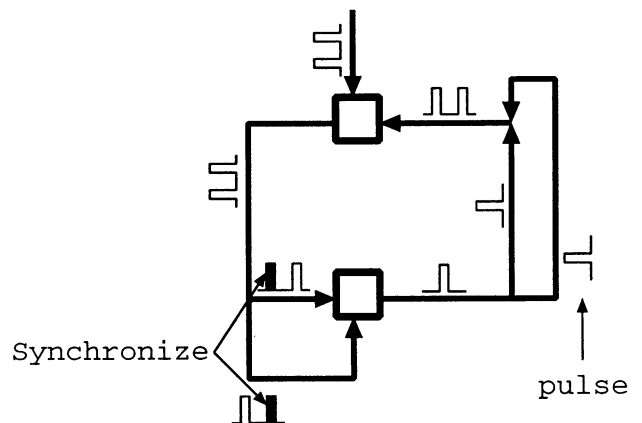
Figure 6: A loop that accept a specific signal pattern

Overcrowding pulses can be eliminated by adding short length inhibitatory loops (Fig. 7. In that case

once the neuron fires, following firings are suppressed for a while.

Inhibitatory loops
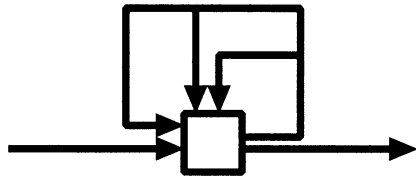
Figure 7: Suppression of continuous firing by inhibitatory loops

Multiple pulse versions can also be considerd. But the operation of such loops may become unclear.

3.3 Feedback loops as autonomous signal generators

Now let's consider more complex loop. Total number of pulses in a loop increases when one pulse is splitted in a branch of axons, and decreases when accumulate of a neuron counter occurs. For simplicity, we don't take account incoming pulses. If we can ballance those increment and decrement rate well by shaping network structure, signal trains may remain long time changing their form.

It is still unclear that the genetic process can evolve such an structure. But if it is possible, such loop works as signal generators. And the remaining network in CAM-brain can take signal train from the loop and adapt to it. In this case, the loop can be considered as independent autonomous existance although it is internal of the CAM-brain network. If signal train from such loop is similar to a certain inputs from real outside world, adaptation to the existance should be considered a "imaginary training"

4 Conclusions

This paper proposed to see certain feedback loops in the CAM-Brain network as structures to store signal trains for on-line adaptation in a slow environment. The proposed structures are very natural from the graph theoretical view point and expected to be easily evolved using genetic process. It was also shown that signal generating feedback loops may be regarded as miniature of external environment for CAM-Brain.

References

[1] de Garis, H. "CAM-BRAIN: The Evolutionary Engineering of a Billion Neuron Artificial Brain by 2001 which Grows/Evolves at Electronic Speed Inside a Cellular Automata Machine (CAM)", in *"Towards Evolvable Hardware"*, Springer, Berlin Heidelberg, NY, 1996.

[2] Gers, F. and de Garis, H. "CAM-Brain: A New Model for ATR's Cellular Automata Based Artificial Brain Project", in *"Evolvable Systems: From Biology to Hardware"*, Springer, Berlin Heidelberg, NY, 1997.

[3] Gers, F., de Garis, H., and Korkin, M. "CoDi-1bit: A Simplified Cellular Automata Based Neuron Model", Artificial Evolution Conf. (AE97), Nimes, France, 1997.

[4] Korkin, M., de Garis, H., Gers, F., and Hemmi, H. "CBM (CAM-Brain Machine) : A Hardware Tool which Evolves a Neural Net Module in a Fraction of a Second and Runs a Million Neuron Artificial Brain in Real Time", Genetic Programming Conf., Stanford, USA., 1997.

[5] Hemmi, H., Shinozawa, K., Hikage, T., and Shimohara, K. "Learning and Relocation capabilities of CAM-brain machine", AROB 3rd, Beppu, Japan, 1998.

[6] Fukushima, K., and Miyake, S. "A self-organizing neural network with a function of associative memory: Feedback-type cognitron", Biol. Cybernetics 12, 204–215, 1978.

[7] Jordan, M.I. "Attractor dynamics and parallelism in a connectionists sequential machine", 8th Conf. of Cognitive Science Society, Amherst, 1986.

Multi-Agent Based Neural Network
as a Dynamical Brain Model

Shinya OHTANI

Graduate School of Infomatics, Kyoto University
Sakyo Kyoto Kyoto
ATR Human Information Processing
Research Laboratories
Hikaridai Seika Soraku Kyoto
ATR Media Integration & Communications
Research Laboratories
Hikaridai Seika Soraku Kyoto

Masahiro ISHIDO

Department of Computer and System Engineering
Faculty of Engineering, Kobe University
Rokkodai Nada Kobe Hyogo

Katsunori SHIMOHARA

Graduate School of Infomatics, Kyoto University
Sakyo Kyoto Kyoto
ATR Human Information Processing
Research Laboratories
Hikaridai Seika Soraku Kyoto

Abstract

This paper proposes a new Neural Network model composed as an evolutionary multi-agent system. In this model, neurons function as agents, and they can grow, move, self-replicate, and evolve. The structure of the network is self-organized by such behaviors of the agents and their interactions. To verify its functionality, simulations are performed in which the model is applied to sound classification and the results show that the model can learn to distinguish sine and square waves.

Keywords: evolutionary neuron, multi-agent, MAB-Net, sound recognition

1 Introduction

A lot of studies on Artificial Neural Networks (ANNs) have been conducted and applied to various applications, ever since they were proposed as information processing mechanisms based on various brain models [1~3]. Almost all ANNs model neural plasticity to achieve learnability, but employ static structures that do not change from the first state to the end state in the learning process. Static structures do make it easy for us to control the learning process, but they may impose some restrictions on the freedom of learnability. If a robot uses a conventional ANN, for example, it still seems hard for the robot to learn flexibility and adaptability like a human being.

In order to overcome this drawback, we attempted to introduce a flexible and adaptable structure into a Neural Network model in this research. This paper proposes a new Neural Network model composed as an evolutionary multi-agent system --- the MAB-Net (Multi-Agent Based neural Network) model. In this model, neurons function as agents, and they can grow, move, self-replicate, and evolve. Accordingly, the structure of the network is self-organized by such behaviors of the agents and their interactions. In the MAB-Net model, learning is achieved by the neurons' (=agents') behaviors such as cell division, growth, movement and so on in a two-dimensional space, and their interactions.

In other words, the MAB-Net model is based on a concept that regards the brain as a growing and evolving system, or which emphasizes the importance of dynamical processes in the brain. In this model, a neural network grows using genetic information and evolves autonomously like the CAM-Brain model [4].

In the CAM-Brain model, the whole system is generated as one individual after the sufficient growth of all neurons. Therefore, a longer number of neurons denotes a greater computational cost to execute sufficient genetic

manipulation for the neural network.

To overcome this problem, the MAB-Net model employs cell-division and the evolution of neurons every time in its running process, instead of repeating the growing phase and genetic manipulations like the CAM-Brain model. In this paper, the MAB-Net is introduced and investigated on its applicability for sound recognition to verify the effectiveness of this model.

2 Multi-Agent Based Neural Network (MAB-Net)

2.1 Overview

The MAB-Net model proposed here has two sides in one model, i.e., a neural network and an evolutionary multi-agent system. In this model, a neuron corresponds to an agent.

Figure 1 shows an image of the MAB-Net as a neural net with some neurons. Every neuron has a nucleus, represented by a black square, and a body surrounding the nucleus, represented by a dark gray part, as shown in Figure 1. The light gray parts denote the dead bodies of neurons. Neurons whose bodies contact each other have connections to transmit signals.

An agent corresponding to a neuron plays a role in growing the neuron's body, moving its nucleus, and/or self-replicating/dividing its nucleus and body, etc. That is, all agents work to generate, grow, and evolve the structure of MAB-Net as a neural network through their interactions.
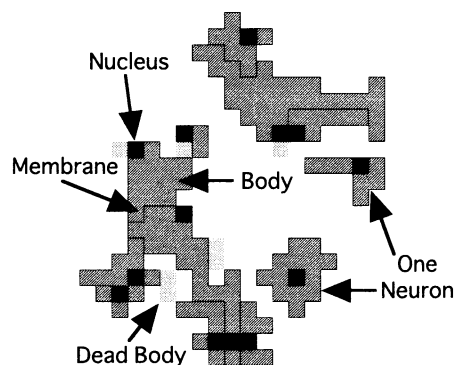
Figure 1: **Image of MAB-Net.**

2.1.1 Behaviors as a Neural Network

Figure 2 shows the neural network described in the usual manner --- the translated version of the MAB-Net in

Figure 1, to explain its behaviors as a neural network. Neurons are represented by circles, the connections between them by lines, and the connection weights by dots in Figure 2. There is no distinction between axons and dendrites, because the neurons are connected through their contacting bodies. By considering the lengths of their contacting faces as synapses, however, we can define the connection weights there.

Accordingly, the MAB-Net can function as a neural network similar to the conventional ANN. Each neuron has an internal potential, fires when its potential summed-up weighted input from connected neurons reaches some threshold level, and transmits its nervous excitement to connected neurons. The nervous excitement transmissions of all neurons are executed at the same time.
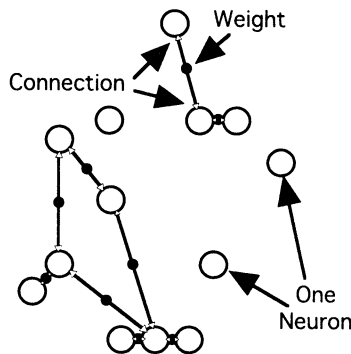
Figure 2: **Neural network corresponding to the MAB-Net in Figure 1.**

2.1.2 Neurons as Evolutionary Agents

In the MAB-Net model, neurons also function as agents. In this point of view, the agents have no relationship to the nervous excitement transmissions, but can change the structure of the neural network dynamically.

Each agent has a gene and some energy. The gene works as a strategic program, and the energy is consumed when the agent executes its strategic program. If the energy runs out, the agent will die. We have introduced energy into this model to decrease agents, which can increase if they have the Divide commands; energy should be kept not to exterminate the whole system.

We prepared the following seven types of behaviors for agents so that they can generate, grow, and evolve the structure of a neural network through their interactions.

(1) Movement   (Move(ω))
  The nucleus (the reference point of the behaviors of an agent) moves in the body to direction ω from the standard angle, only when the direction exist in the body.

(2) Change of direction   (Turn(ω))
  The agent rotates its standard angle ω degrees.

(3) Growth   (Grow(ω))
  The agent grows in direction ω from the standard angle from the nucleus, only when the space in the direction is vacant or is a dead body area. In the latter case, the growing agent can get special energy.

(4) Absorption   (Absorb(ω))
  The agent absorbs its contacting part to the neighboring agent

in direction ω from the standard angle from the nucleus, when the body of the neighboring agent is in this direction. If this command works, the attacking agent gets energy, and the attacked agent loses its part. If the lost part is the nucleus, the attacked agent dies.

(5) Division   (Divide(ω)))
  The agent divides itself into two agents in its body. First, it makes a copy of the nucleus in direction ω from the standard angle from the original nucleus, only when its body is in the direction. The gene and other information like the standard direction of the nucleus are also copied in this step with some copy errors working as mutations. Next, the new and old nucleuses make their own cell membranes by lengthening straight the boundary between them to the original body, and then work as individual agents.

(6) Memory   (Memory(D))
  The agent rewrites the new value D in its memory, which effects its selection of the next behavior at a branch instruction (Jump) in the strategic program.

(7) Do nothing   (Wait)
  The agent waits, and its energy is saved.

The interaction commands, Grow and Absorb, are related to the dynamic growth of the neural network structure. The Divide command and the death of an agent are related to the change of the structure itself. The Memory command enables an agent to memorize some event slightly so that the agent can use this information as a long-term memory as well as a short-tem memory with its internal potential.

2.1.3 Genes of Agents

Each gene as a strategic program is composed of fifteen types of command codes not directly related to the seven behaviors described above. The behaviors and command codes are mapped as shown in Table 1. The number of behaviors is less than the number of command codes, after taking the relationship between the numbers of condons and amino acids of real biological cells.

An example of the strategic program is shown in Figure 3. Each strategic program is composed of fifteen program modules in this paper. The modules are layout easy to grasp relations between them in Figure 3. Each module is composed of a string of the command codes, and the top command code of each string is the same as the module number.

2.2 Agent's Behaviors and Interactions

An agent behaves according to its strategic program, which is executed as follows: The first time, a module is chosen at random, and the command codes of the module are executed from the top one after another until the end of the module. If a jump code is executed, the module indicated by the number of the jump code is chosen. If the jump code has plural numbers as jumping destinations, one of them is selected according to some rules based on the situation of the neuron and/or the agent. If the executing module does not have any jump code, a command error is output, and then another module is chosen at random. As an example here, the behaviors of the agent (A_1),

Table.1 Command codes and behaviors.

Command code	Behavior type
1	Grow
2	Move
3	Turn
4	Absorb
5	Divide
6	Memory
7	Wait
8	Grow
9	Move
10	Move
11	Divide
12	Grow
13	Divide
14	Grow
15	Grow

Figure 3: **Strategic program of an agent.**

whose gene is displayed in Figure 3, are traced in the following. Figure 4 shows the growth process and death related to the neuron corresponding to the agent (A_1). That is, the process itself in Figure 4 is the trace of the agent's behaviors. In Figure 4, the black square with a number indicates the nucleus, and the number indicates the agent's number. Accordingly, different numbers represent different neurons and agents. The dark gray part indicates the body of the neuron, and the light gray parts indicate the dead bodies of agents that lost energy or were absorbed.

Figure 4: **Behaviors of agents through the growing process of neurons.**

First, A_1 executes the top command of Module 11 (M_{11}) chosen at random. A_1 divides itself and makes A_2 at this time because the top command of M_{11} is a Divide code. In this example, the strategic program of A_2 and its executing point are generated exactly the same as A_1. Now, the mutation rate is set to 0 for simplicity, but if the mutation rate is not 0, a different gene is generated by bit copy errors. Next, A_2 is added to the execution queue, and A_1 and A_2 execute their strategic programs one after the other.

After A_1 executes M_{11}, the executing point jumps to M_2. After A_1 executes M_2, the executing point jumps to M_3 chosen at random.

In this example, the number of neurons (=agents) increased to eight by cell division, and finally only A_8 remained by the neurons absorbing each other; not shown in Fig. 4. Then, A_8 fell into an endless loop of $M_3 \rightarrow M_{13} \rightarrow M_7 \rightarrow M_5 \rightarrow M_3$ to make its copy and eat the copy repeatedly.

3 Simulation

In order to verify the functionality of the MAB-Net model, we applied the MAB-Net model for a simple sound classification problem to categorize sound inputs into two sounds: sine and square waves.

3.1 Experimental Design and Procedure

3.1.1 Procedure

The procedure for this simulation consisted of the following three steps: (1) the generating step, (2) the learning step, and (3) the testing step.

In the first step, agents generated and grew the structure of a neural network, i.e., the MAB-Net, based on their genes as strategic programs and the interactions between them.

In the second step, the MAB-Net was trained so that it could generate a correct answer as its output to an input. Every time input sound was given to the MAB-Net, its output was calculated by counting the number of firing neurons and their times, and mapped on a space represented by the firing rate and the times of neurons for some period. Therefore, the judging line to separate the output space into two spaces should be adjusted so that the separated space can properly correspond to the two types of sounds, as shown in Figure 5.

The third step was to investigate the performance of the MAB-Net, and the correctness rates for several trials were calculated according to the judging line.

The details on how to deal with inputs/outputs in the MAB-Net and how to carry out learning are described in the following sections.

3.1.2 Sound Input to the MAB-Net and Output

Since the MAB-Net does not have a specific input layer, we employed a special approach to deal with inputs,

Output Judging Line

Output of "Sine wave"

Output of "Square wave"

---------- Out put when sine wave inputted

——— Out put when square wave inputted

Firing Rate

0   100 [%]
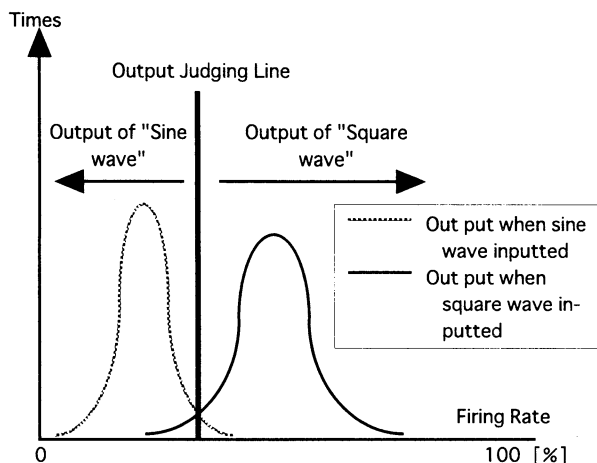
Figure 5: **Image of Output Space and Judging.**

which were never used in conventional ANNs. The image of this unique approach is shown in Figure 6. Assuming a virtual flat space on which neurons exist, a speaker is located above the space. The speaker emits sound signals to neurons on the space. Some amount of potential, which is proportional to the amplitude of the sound wave, is added to the internal potential of all neurons. Ideally, the MAB-Net can deal with an input in proportional speed to the sampling rate of the sound, and carry out nervous computation as well, however, this is impossibility. Therefore, when neurons transmit their activations to neighbors, the above process for sound input is executed. That is, what the MAB-Net deals with is a sparse sound wave, not a continuous wave.
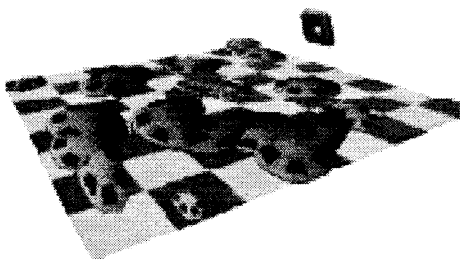
Figure 6: **Image on how to get an input.**

3.1.3 Learning in the MAB-Net

As preparation for learning in the MAB-Net, neurons on the virtual flat space are divided into two groups, based on their positions on the space, for example. The two groups are respectively related to the two output spaces separated by the judging line mentioned above.

One of the two sounds is chosen at random every two seconds, and is given to neurons as input. All of the outputs of all neurons are checked whether they are fired or not, and the number of all firing neurons for every two seconds is summed up. If a neuron fires when dealing with one of the sound inputs, and if the neuron is predetermined to fire for the sound input, the neuron gets energy. the MAB-Net does not learn directly to acquire an ability to recognize sounds; however, the MAB-Net can acquire that ability by having agents cooperate with each other not to die out.

3.2 Simulation Results

We conducted simulations on sound classification involving sine and square waves in the following two cases; (1) as shown in Fig. 5, the left and right sides of the output space are respectively allocated to sine and square waves, and (2) as the opposite case to case (1), the allocation between spaces and waves is reversed.

The average of the correct rates, in the results of the simulations, for case 1 was 89.6%, and it was 91.6% for case 2. The time spent for learning was less than several hundred seconds, using a Power PC604/150.

In spite of only sparse sound inputs being used, the acquired performance was satisfactory. In particular, concerning the time needed for learning, the MAB-Net was found to work much better than general genetic algorithms which often spend many hours.

4 Conclusions

We have proposed a Multi-Agent Based neural Network (MAB-Net) model that has two sides; i.e., a neural network and an evolutionary multi-agent system. In this model a neuron is related to an agent, and the agents' behaviors and their interactions function to generate, grow, and evolve the structure of the MAB-Net as a neural network. The MAB-Net model is much closer to the brain than conventional artificial neural networks in terms of its degree of freedom for forming the structure of a neural network.

We applied the model to a sound classification problem to verify functionality, and found that the model works much better than general genetic algorithms in the speed of learning.

Out future plans are to first analyze of the current results and then to investigate the reason why the MAB-Net model gives a good performance.

References

[1] T. Yahagi, *Neural Network and Fuzzy Signal Processing*, CORONA PUBLISHING CO., LTD. , (1998)

[2] K. Itoh, *Biology of the Brain and Nerve*, BAIHUKAN, (1975), pp. 88-110

[3] K. Aihara, *Neural Computer*, Tokyo Electric University Publisher, (1988)

[4] ATR Human Information Processing Evolutionary System Laboratory, *Artificial Life and Evolutionary System*, Tokyo Electric University Publisher, (1998)

[5] K. Shimohara, *Artificial Life and Evolutionary Computer*, KOGYO TYOSA KAI, (1998)

[6] K. Ueda, K. Shimohara, and H. Iba, *Method of Artificial Life*, KOGYO TYOSA KAI, (1995)

[7] H. Kitano, *Genetic Algorithm*, SANGYO TOSHO, (1993), pp. 235-262

Does The Butterfly Effect Take Place in Human Working Memory?

Andrzej Buller[1,2] and Katsunori Shimohara[1,3]

[1]ATR Human Information Processing Research Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan

[2]Faculty of Electronics, Telecommunications and Informatics
Technical University of Gdansk
G.Narutowicza 11/12, 80-952 Gdansk, Poland
buller@pg.gda.pl

[3]NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
katsu@cslab.kecl.ntt.co.jp

Abstract

Why people hesitate to do something or not to do even when the data available to them remain constant? The neural model of the working memory (WM) we present explains hesitation as an emergent property of a complex dynamic structure of information stored/processed in WM which is considered as a space inhabited by the 'society' of memes. A large population of identical memes can cause a feeling, judgment or intention of an individual. The memes navigate all over WM and interact with one another in the way resembling genetic crossover. Since new facts born at several places in WM may contradict each other, different populations of contradictory feelings, judgments and plans grow and try to dominate over WM. A simulation of the process shows that WM's state goes to an attractor, however, in some cases it is a two-foci 'strange' attractor. Hence, sudden mental shifts, as, say, from love to hate and back from hate to love, may be caused by minute fluctuations of densities of streams of memes entering WM. The complex system theory knows this phenomenon as the "butterfly effect". We argue that this effect takes place in human or animal mind and can take place in an advanced robot.

Keywords: working memory, memes, complexity, social judgment, artificial neural networks.

1 Introduction

Every person, except mentally ill ones, judges autonomously, hence, we cannot perfectly predict individuals' behaviors [18]. Moreover, there is empirical evidence that social judgment shows temporal variations even in the absence of new information or social stimuli [16][22]. Building a psychologically plausible model of mind, one should remember, that the information processed in the brain is often incomplete and full of fuzzy notions, as well as contradictory statements. We present a neural model which (1) explains hesitation as an emergent property of a complex dynamic structure of information stored/processed in the working memory (WM), (2) copes with fuzzy, incomplete and contradictory data, and (3) is formulated in terms of "Society of Mind" and "memes". A usefulness of a cognitive model depends on possibilities of its implementation, hence, we consider its implementation as a large-scale neural network to be evolved in a cellular automaton built in the framework of ATR's CAM-Brain Project [13].

Working memory (WM) allows us to keep active 5-9 chunks of information for a few seconds. It may play a key role in higher cognitive processes regulating the flow of information during categorization, planning, reasoning, problem solving etc. [2]. In the ACT-R—seemingly the best elaborated cognitive model—WM is a "blackboard" from which productions read information and to which they write conclusions [1]. In Tulving's taxonomy of memory systems WM works in parallel and seems to integrate information stored in four other kinds of memories [21]. Buller [5] proposed the "4 + 1 Memory Model" which is considered as the target structure of an artificial brain being built in the framework of the ATR's CAM-Brain Project [8].

Fuzziness of notions seems to be a natural consequence of the way we categorize perceived objects. As Eleanor Rosch [18] experimentally confirmed, people judge objects to be members of a particular category to different degrees. This observation fits well the earlier concept of fuzzy sets [23]. In order to employ the notion of fuzzy set in large-scale brain-like neural networks, a non-symbolic representation of fuzziness has been proposed [3][4]. One of recent solutions employs a frequency-based representation of fuzzy notions and assumes the possibility that in the working memory contradictory statements may co-exist [6].

Meme (term coined by Richard Dawkins) according to Plotkin [17 p.251] is the unit of cultural heredity analogous to gene. Brodie [9] sees a meme as a unit of information in a mind whose existence influences events such that more copies of itself get created in other minds. In order to avoid confusing memes with their expressions, Buller [7] defines memes as "units of cerebral code representing signals, or words, or sentences, or rules, or plans, or feelings, or verbal or non-verbal ideas, which interacts with each other in the course analogous to genetic interactions. Such view of memes provides a united framework for a synthesis of a model of the mind seen as a society of interacting agents [15], compatible with Calvin's proposal of a hexagon-based neural workspace in which populations of memes grow and fight against each other for domination in the workspace [10].

2 Modular Working Memory

In the presented model of mind, the working memory (WM) is a space inhabited by a society of memes representing feelings, judgments and plans. The memes navigate all over the space and interact with one another. Meme interaction consists in an exchange of parts of their informational contents in a way resembling genetic crossover. As we show, such interactions can explain an individual's hesitation in face of necessity to make a decision in a specific social situation.

2.1 Example of social behavior

Let us consider a female individual determined to follow the rule: "I can have a date only with a guy who is nice and rich". The rule can take the form "I can have a date with X, if X is nice and X is rich". Now let us assume that she meets somebody who proposes her a date, but there is no visible clues about his richness or poverty, while, according to her criteria, only 70% of the features taken into account let him to be labeled 'nice'. She hesitates. "To agree or not to agree?" The roots of the phenomenon still remain an open question. The cognitive model enables us to explain this in terms of meme interaction.

Let us assume that in face of cues causing contradictory conclusions, a sensorium of the individual, in cooperation with a semantic memory, produces streams of memes representing contradictory data and directs them to her WM. Since 70% of the dater's features qualifies him as "nice", a number of memes representing the assertion "[the dater is] nice" and arriving to WM in a period of time will be 7/3 times bigger than the number of memes representing the assertion [the dater is] not nice" arriving in the same time. No clues about the dater's richness or poverty causes that a stream of the memes representing "[the dater is] rich" and [the dater is] not rich" will be equal. Gradually WM is getting full of contradictory memes.

2.2 Meme interaction

Let us imagine WM as a workspace resembling a honeycomb—a structure consisting of hexagonal tiles. Let us assume that each meme can, jumping from one tile to another, navigate all over the workspace. Let us consider a simple model in which only two memes can occupy a single tile, and the rule of meme traffic is such that all meme movement vectors must have non-zero positive horizontal projection. The rule protects the society of memes from conflicts on the ground who is to enter a given half-tile when two memes want. Memes who meet in a cell interact. The interaction is equal to an exchange of informational content.

Every meme is a sequence of three patterns denoted here as characters. If only the second character is non-empty, the meme represents a fact. The first and the third characters are conditions making the meme a representation of a rule in which the second character is a conclusion. Let us assume that **ORO** = "rich", **OrO** = "not rich", **ONO** = "nice", **OnO** = "not nice", and **NAR** = "Agree [for the date] if [the dater] is nice and rich". The most important kind of interaction is local production: a navigating meme, representing a fact, meets a meme representing a related rule, matches it, and then, as a result of the union, a new fact or a reduced rule is born. For example: **ORO** meets **NAR**, matches it, and the reduced rule **NAO** appears (see Fig. 1). The interaction is equal to an exchange of single characters, which resembles the genetic crossing-over. A meeting of the rule **NAR** with the fact **OrO** ("not rich") results in an immediate appearance of the resulting fact **OaO** ("Don't agree") since in face of evidence of the dater's poverty his niceness has no significance.
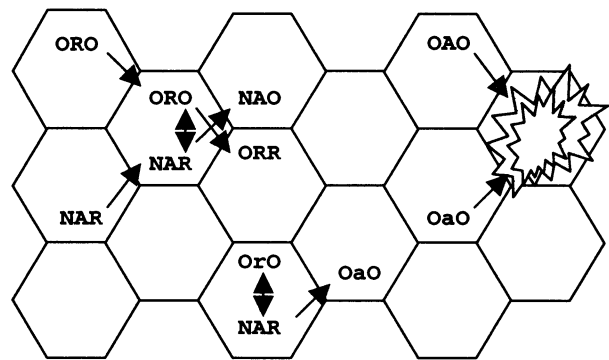
Fig. 1. Meme interaction in a tile-based working memory. A meme **ORO** ("rich") meets **NAR** ("Agree [for the date] if [the dater] is nice and rich"), matches it, and the reduced rule **NAO** ("Agree if nice") appears. A meeting of **NAR** with **OrO** ("not rich") results· in an immediate appearance of the resulting fact **OaO** ("Don't agree"). When contradictory facts **OAO** ("Agree") and **OaO** ("Don't agree") meet, they annihilate.

When contradictory facts, as for example **OAO** ("Agree") and **OaO** ("Don't agree"), meet, they annihilate. Since the same happens at the same moment at several WM's places, and particular fact and rules exist in multiple copies, WM becomes a "war theater" in which the populations of contradictory feelings, judgments and plans fight for survival.

2.3. Plan generation and decision making

The presented idea of WM seems to be suitable as an explanation how people work out their plans. For the simplified model a three-character meme may represent a three-step plan. From a small set of randomly generated sequences, through meme interactions a population of best plans may dominate WM. To implement this an external evaluating device (based on a model an environment and simulation of its changes when a given plan is executed) should be proposed.

The final individual's decision is based on changes of WM's state for a certain period of time. The state is assumed to be a point (X, dX) located in a 2D phase-space, where X is the percentage of memes carrying copies of a given resulting statement $\langle Q \rangle$ amongst memes carrying copies of the statement $\langle Q \rangle$ or $\langle q \rangle$ ("not Q"), while dX is assumed to be X(t) - X(t-1). Simulation of the process shows that the state, depending on the perceived data, goes to an appropriate attractor. In typical situations the atractor is either the point (0, 0) or (100, 0). However, it has been observed, that in some cases the state goes to a two-foci 'strange attractor' which means that the agent "hesitates" what to do. This phenomenon makes the model psychologically plausible. Fig. 2. shows the most typical behaviors of the model of WM based on square tiles [7].

Why, even in case of constant average density of streams of memes arriving to the modeled working memory, the modeled mind hesitates? Buller [7] strongly suggests, that the "butterfly effect" known from the theory of complex systems [11] may take place in such kind of WM. Even when average density of streams of arriving memes is constant for a long period of time, this does not mean that short-term fluctuation of the density cannot take place. If the complex system of meme distribution in the working memory is the subject of the "butterfly effect", then even small fluctuations of the density of streams of arriving memes can prevent the system from fixing itself in a particular point of the phase space.

3 Implementation

An extreme computational power is necessary to build a full working space for a society of memes. Such a power can be attainable owing to CoDi ("Collect and Distribute") technique combining neural engineering with genetic programming and cellular automata

paradigm [14]. Every tile of WM can be implemented as a circuit consisting of one or more CoDi modules.

A CoDi module is a cube consisting of 24 × 24 × 24 3D Cellular Automata cells. A chromosome containing growth instructions is used to grow a neural network inside the module, which means that some of the cells form neuron bodies, dendrites and axons. 1-bit signals are collected by dendritic cells and directed to a relevant neuron-body cell. When certain conditions are satisfied, a 1-bit signal is generated and, using axons, delivered to other neurons. Inside a single CoDi module up to 1152 neurons may be grown.

A desired CoDi module can be obtained using a genetic algorithm. A special purpose supercomputer "CBM" (Cellular [Automata-based] Brain Machine) is being built in the framework of ATR's CAM-Brain Project [12][20]. The CBM can update a 64,640-module structure about 150 times a second, which gives a computational power sufficient to control in real time a life-sized animal, as well as several kinds of memory models [13].
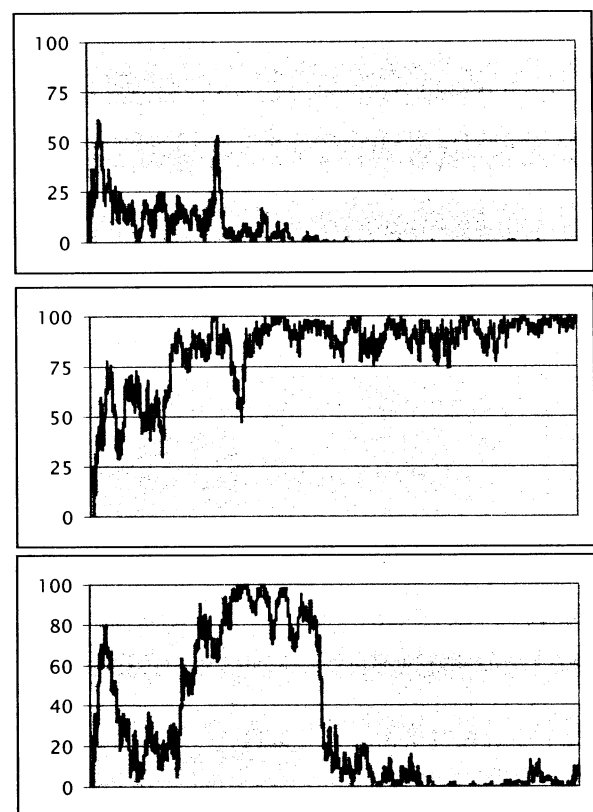
Fig. 2. Changing state of the modeled working memory. The state is assumed to concern a particular judgment (100 – extremely positive, 0 – extremely negative). Since in all three cases the informational grounds for the judgment remained constant, we explain the observed 'hesitations' as a kind of the 'butterfly effect'.

4 Concluding remarks

We proposed a model of the working memory as a space populated by society of memes. The results of simulation of the memory show that a "debate" in the society of memes can be equal to a massively parallel fuzzy inferencing from contradictory statements. We assert that the phenomenon of hesitation observed during simulation of the process is the "butterfly effect" observed in other complex systems. If the idea is close to a correct model of human mind, it must be taken into account that artificial brain-like systems, build using such and equipment as the described CBM, may demonstrate human-like limited predictability, their social judgments and decisions may appear in their working memories as emergent phenomena.

References

[1] Anderson JR (1993) *Rules of the Mind*, Lawrence Erlbaum: Hillsdale, NJ.

[2] Baddeley AD (1999) *Essentials of Human Memory*, Psychology Press: Hove.

[3] Buller A (1995) Operations on Multimodal Records: Towards a Computational Cognitive Linguistics, *Technical Report TR-95-027*, International Computer Science Institute, Berkeley.

[4] Buller A (1996) Fizzy-Fuzzy Inferencing, In: Furuhashi T & Uchikawa Y (eds.) *Fuzzy Logic, Neural Networks, and Evolutionary Computation*, Springer: Berlin, 172-187.

[5] Buller A (1998) [*Artificial Brain – No Longer Science Fiction*] (in Polish), Prószynski i S-ka: Warszawa.

[6] Buller A (1999) Fuzziness, mega-networks and CoDi technique, *4th Conference on Neural Networks and Their Applications, May 18-22, Zakopane, Poland*.

[7] Buller A (to appear) *Self-Organization of Mind*, Ph.D. Dissertation, Department of Psychology, Warsaw University.

[8] Buller A, de Garis H (1998) Brain-building strategy: some remarks and questions, *Proceedings of the Workshop on Intelligent Information Systems (IIS'98)*, Malbork, 188-193.

[9] Brodie R (1996) *Virus of the Mind: The New Science of the Meme*, Integral Press: Seattle.

[10] Calvin WH (1996) *The Cerebral Code: Thinking a Thought in the Mosaic of the Mind*, A Bradford Book/The MIT Press: Cambridge, Mass.

[11] Casti JL (1994) *Complexification : Explaining a Paradoxical World Through the Science of Surprise*, HarperCollins: New York.

[12] de Garis H, Gers F, Korkin M, Agah A, Nawa N (1998) Building an Artificial Brain using an FPGA-based "CAM-Brain Machine", *Proceedings of Thirs International Symposium on Artificial Life and Robotics (AROB III'98)*, Oita, Japan, 258-262.

[13] de Garis , Buller A, Korkin M, Gers F, Nawa E & Hough M (1999) ATR's Artificial Brain ("CAN-Brain") Project: A Sample of What Individual "CoDi-1Bit" Mo Vallacher RR & Nowak A (1994) del Evolved Neural Net Modules Can Do With Digital and Analog I/O, *The First NASA/DoD Workshop on Evolvable Hardware*, Pasadena, California, 102-110.

[14] Gers F, de Garis H, Korkin M (1997) CoDi-1Bit: A Simplified Cellular Automata Based Neuron Model, *Evolution Artificielle (AE'97)*, Nimes, France, 211-229.

[15] Minsky M (1987) *The Society of Mind*, Simon & Schuster: New York.

[16] Nowak A & Vallacher R (1998) *Dynamical Social Psychology*, Guilford Press.

[17] Plotkin H (1993) *Darwin Machines and the Nature of Knowledge*, Harvard University Press: Cambridge, Mass.

[18] Rosch E (1975) Cognitive representations of semantic categories, *Journal of Experimental Psychology: General, 104*, 192-223.

[19] Shimohara K (1999) Evolutionary Systems for Brain Communication, *Proceedings of International Symposium on Low-Power and High-Speed Chips (COOL Chips II)*, Kyoto, Japan, 37-50.

[20] Shimohara K & Hemmi H (1999) [Evolving Artificial Brain] (in Japanese), *Computer Today*, 4, 4-9.

[21] Tulving E (1995) Organization of Memory: Quo Vadis?, In: Gazzaniga MS (ed.) *The Cognitive Neurosciences*, A Bradford Book/The MIT Press: Cambridge, 839-847.

[22] Vallacher RR & Nowak A (1994) The Stream of Social Judgment, in: Vallacher RR & Nowak A (eds.) *Dynamical Systems in Social Psychology*, Acasdemic Press: San Diego, 251-277.

[23] Zadeh L (1965) Fuzzy Sets, *Information and Control*, 8, 338-353.

Axonal Growth in Evolutionary Neurogenesis

P. Eggenberger

ATR Human Information Processing Research Laboratories

2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02 Japan

eggen@hip.atr.co.jp

Abstract

Creating methods to evolve neural networks is an important goal in the field of artificial evolution. The main problem is how to encode the structure and properties of the neural network in the genome. If one overloads the genome with detailed information for a network the evolutionary time increases prohibitively. If the genome is too simple, only simple problems can be solved. Since Nature has found an efficient and evolvable solution to this problem, it is worthwhile imitating the mechanisms on how biological neural nets are generated. In this paper a model is proposed in which artificial genomes tune the ability of axons to find, detect and connect to specific targets. Initial simulation results of simple tasks are evolved and the genetic tuning of the developmental processes for artificial evolution is discussed.

Keywords: *axon, receptor concept, evolution*

1 Introduction

The feat of information-processing in brains are determined to a large amount by the structure of the network of connections between cells [1, 2]. In animals with bilateral symmetry, the information exchange between the two halves of the body is controlled by neurons, which send their axons across the midline of the central neural system [3, 4]. These axons move along trajectories or commissures, which can be found already in primitive, bilateral organisms. Noting that these mechanisms of connecting bilateral regions are evolutionary old and that they are functionally important, a model is proposed in which mechanisms of axonal behavior can be evolved. Therefore, one can investigate the relationship between an artificial genome, the properties of axons and the resulting structure of neural network. The objective of this paper is to simulate the principles of axonal outgrowth, to understand their evolutionary advantages and to apply them to artificial evolution in order to get better evolutionary techniques. All the properties of the neurons and axons are encoded in an artificial genome and can be changed either by hand or artificial evolutionary techniques such as evolutionary algorithms. The axons' behavior is influenced by the expression of receptors on its surface and by artificial chemical substances from other cells or sources. It is possible to inhibit or express certain genes, which change the properties of the axons and change accordingly their behavior.

Gruau and Whiteley [5] encoded the developmental process as a grammar tree. In this approach the cells inherit their connections and no context sensitive development is possible. Fleischer and Barr [6] used a genetic encoding base on if-clauses to allow for differential gene expression. Vaario and Shimohara [7] proposed a grammar-based simulation tool to control neural growth. Nolfi [8] proposed a developmental model for neural networks based on cell division and cell migration. Dellaert and Beer [9] used a model based on Boolean networks to evolve autonomous agents with developmental processes.

2 Method

Every cell of the model contains a genome in which every gene is controlled by several regulatory units, which act like switches to turn a gene on or off . A gene can be activated or inhibited by artificial signaling molecules. The activity of a gene is calculated according the following formula:

$$A_i(conc_{SM_0}, .., conc_{SM_m})$$
$$= \frac{1}{2}\beta_i * (1.0 + \tanh(\alpha * (\sum_{j=0}(\Theta(a_{ij} * conc_{SM_j} - \theta_j)))))$$

$$\Theta(x) = \left\{ \begin{array}{lll} 1.0 & : & if \ x > 0 \\ 0.0 & : & otherwise \end{array} \right.$$

The above variables have the following meaning:

A_i is the activity of the gene, β_i amplitude of the activity, $tanh(x)$ is the tangens hyperbolicus functions, a_{ij} is the affinity to encode the effect between the regulatory unit i and the transcription factor j, concentration $conc_{SM_j}$ of the signaling molecule j, θ_j is a threshold value.

The specific values of all parameters are assigned by an evolutionary strategyand they change during the simulated evolution.

2.1 The Neuron

In this paper only the properties of the axons and their partners to build connections are implemented. The functional properties, already used discussed in [10], are neglected. In Figure 1 a typical neuron with its axon is illustrated. Every neuron is potentially able to emit an axon. It will do so, if and only if a gene controlling the outgrowth of axons is active. The neuron contains a genome and controls the expression of axonal receptors.
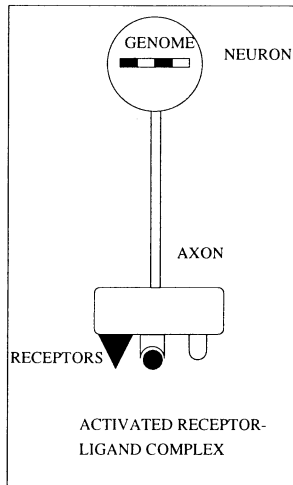
Figure 1: Illustration of a typical neuron. Every neuron contains an artificial genome, which controls the properties of the cells. Depending on the gene activities different types of receptors are expressed, which detect specific signaling molecules. The specific interactions between the receptors and these molecules (the production of receptor-ligand complexes) will be used to decide in which direction the axon will move.

2.1.1 Axonal Growth

The interactions between these receptors and the signaling molecules, determine the direction of the next move the axon will undertake. The effect of the signaling molecules on an axon at a certain place is calculated as follows:

$$I_i = \sum_{i=1}^{n} \omega_{i,j} * r_i * s_j - \theta_i \qquad (1)$$

I_i influence of signaling molecules on the i-th axon r_i concentration of the i-th receptor at position x,y,z s_j concentration of the j-th signaling molecule at position x,y,z $\omega_{i,j}$ the affinity between signaling molecule i and the receptor j

This formula is calculated in a user-specified range around the current location of the axons, and the next point on the grid will be in the direction of the biggest influence. The axons are able to read gradients of chemicals and avoid or approach them by reading the concentration differences locally. If a cell finds another cell, which contains matching receptor(s) on their surface, a link is established and a signal to the outgrowth gene inhibits a further movement of the axon.

3 Results

In contrast to previous work (c.f. [10]) the genes products are restricted to signaling molecules and receptors as well as the functions to axonal outgrowth and synaptogenesis. The details of how the gene can be regulated can be found in [11]. The aim of the simulations was to check the feasibility of the gene regulatory concept to connect neural clusters with other neurons. In the first example two sources of signaling molecules were placed in two corners of a grid with 30x30 points. At one side 30 neurons and at the opposite side 30 target cells were positioned (see Fig 4). The task of the evolutionary process was to evolve axonal growth mechanisms (expression of axonal outgrowth genes, activation of receptor genes, which were able to read the gradients in order to connect to the target cells). The used fitness was simply to count how many neurons were expressing an axon and to check if the synapses were made at the right places. The outcome of the evolutionary process is illustrated in Figures 2 to 4. A second, more complex example is shown in Figure 5, where two neural clusters had to be connected to two different locations.
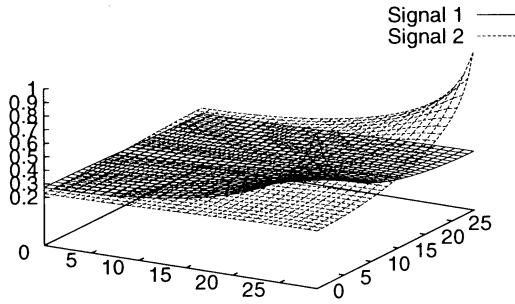
Figure 2: An example of a concentration profile of two different signaling molecules. These signaling molecules are produced by special cells depending on their gene state. (The x-y plane is a grid in which the cells live. The z-axis (0.0-1.0) is an arbitrary chosen concentration unit).
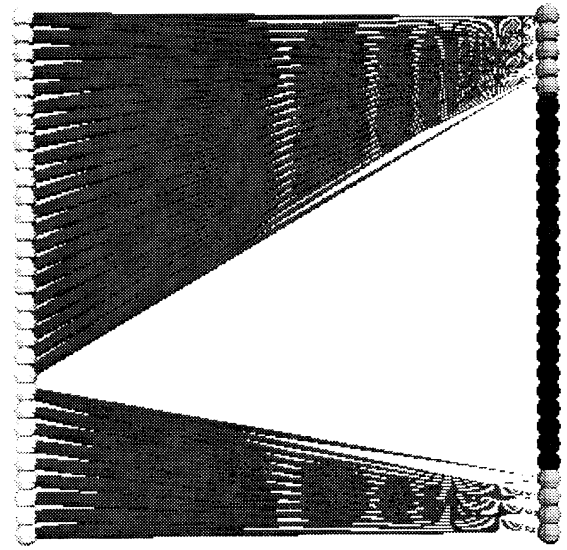
Figure 4: The thirty neurons on the left emitted axons to the right by reading the gradient of the signaling molecules. Note that the asymmetry of connection for the upper and lower side was determined by the asymmetric interpretation of the two different signals.
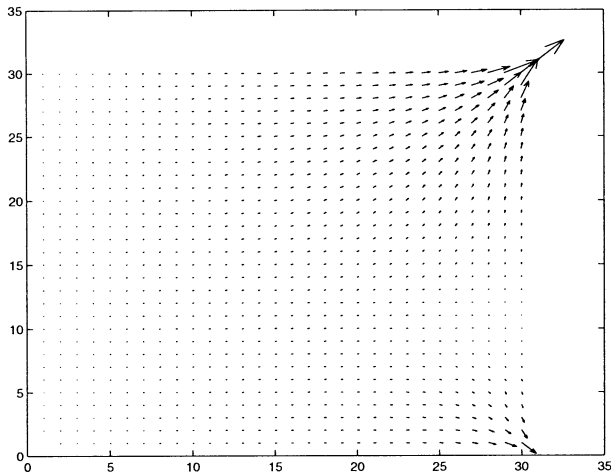
Figure 3: The axon's choice of direction it will move is a function of its expressed receptors and the concentrations of the signaling molecules. These influences on the axons are illustrated as a gradient field. (The numbers of the axis number the locations of the grid cells).
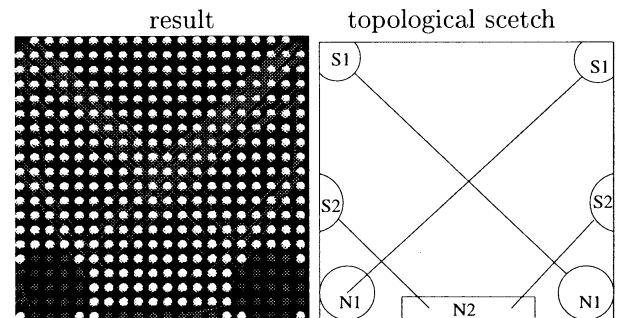
Figure 5: A second more complex example is evolved in which two types of sources (S1, S2) at 4 locations and two types of neurons (N1,N2) were present. The task was to connect the neurons of type N1 with the source S1 in a crossed fashion and the neurons N2 have to be connected to S2. The used grid contained 400 cells, not all of them were neurons.

4 Conclusions

The above preliminary simulations showed that it is possible to build genetically guided axons and to investigate the resulting behaviors by changing the genes. As evolution is a process of ongoing adaptation with necessary change of form and function, those biological mechanisms are selected, which allow such changes without a decrease in fitness [12]. By imitating such evolved mechanisms, it is reasonable to believe that the simulated investigation of these mechanisms allow to ameliorate the actual artificial evolutionary techniques. Already from these preliminary simulations the following principles can be deducted.

1. the developmental mechanisms are independent of the number of cells, in other words, the genetic information will not increase with the number of cells.

2. the receptor concept allows for combinatorial use of a primary set of receptors. By varying the expression of the receptors, many different "guiding modules" can be build, which are easy to evolve.

3. the mechanisms depend not critically on position addresses. Connections are made by dynamic search processes, which allows to change the location of the sources and/or the neurons without decreasing the fitness. Such flexibility is a trait favored for evolution and is useful in evolutionary systems in which the cells can change their places.

Acknowledgments

References

[1] Gerald M. Edelman. *Neural Darwinism.* New York:Basic Books, 1987.

[2] Guilio Tononi, Olaf Sporns, and Gerald Edelman. Reentry and the problem of integrating multiple cortical areas: Simulation of dynamic integration in the visual system. *Cerebral Cortex,* 2:310–335, 1992.

[3] Marc Tessier-Lavigne and Corey S. Goodman. The molecular biology of axon guidance. *Science,* 274:1123–1133, 1996.

[4] W. A. Harris and C.E. Holt. Slit, the midline repellent. *Nature,* 398:462–463, 1999.

[5] Frederic Gruau and D. Whitley. The cellular development of neural networks: The interaction of learning and evolution. Technical Report 93-04, Laboratoire de l'Informatique du Parallélisme, Ecole Normale Supérieure de Lyon, France, 1993.

[6] Kurt Fleischer and Alan H. Barr. A simulation testbed for the study of multicellular development: The multiple mechanisms of morphogenesis. In [13], pages 389–416, 1992.

[7] Jari Vaario and Katsunori Shimohara. Synthesis of developmental and evolutionary modeling of adaptive autonomous agents. In [14], pages 721–726, 1997.

[8] Stefano Nolfi, Dario Floreano, Orazio Miglino, and Francesco Mondada. How to evolve autonomous robots: Different approaches in evolutionary robotics. In P. Meas R. A. Brooks, editor, *Artificial Life IV.* Cambridge, MA: MIT Press, 1994.

[9] Frank Dellaert and Randall D. Beer. A developmental model for the evolution of complete autonomous agents. In [15], pages 393–401, 1996.

[10] Peter Eggenberger. Creation of neural networks based on developmental and evolutionary principles. In [14], pages 337–342, 1997.

[11] Peter Eggenberger and Raja Dravid. An evolutionary approach to pattern formation mechanisms on lepidopteran wings. In [16], pages 337–342, 1999.

[12] J. Gerhart and M. Kirschner. *Cells,. Embryos and Evolution: Toward a cellular and Developmental Understanding of Phenotypic Variation and Evolutionary Adaptability.* Oxford: Blackwell, 1997.

[13] Christopher G. Langton, editor. *Artificial Life III: Proceedings of the Workshop on Artificial Life,* Reading, MA, 1994. Addison-Wesley. Workshop held June, 1992 in Santa Fe, New Mexico.

[14] Wulfram Gerstner, Alain Germond, Martin Hasler, and Jean-Daniel Nicoud (Eds.), editors. *Seventh International Conference of Artificial Neural Networks (ICANN'97),* Cambridge, MA, 1997. Springer.

[15] Pattie Maes, Maja J. Mataric, Jean-Arcady Meyer, Jordan Pollack, and Stewart W. Wilson, editors. *From animals to animats 4: Proceedings of the fourth international conference on simulation of adaptive behavior.* MIT Press, 1996.

[16] Peter J. Angeline, editor. *Congress on Evolutionary Computation,* Piscataway, NJ, 1999. IEEE Neural Network Council. Workshop held July,1999, in Washington DC,USA.

Optimization of Neural Network Structure Based on A Hybrid Method

Masanori Sugisaka and Zhijun Liu

Department of Electrical and Electronic Engineering
Oita University
700 Dannoharu, Oita 870-1192, JAPAN
Phone: +81-975-54-7831, Fax: +81-975-54-7841
msugi@cc.oita-u.ac.jp   zjliu@cc.oita-u.ac.jp

Key word: Neural network, Optimization, Genetic algorithm.

Abstract

In this paper, genetic algorithm (GA) is implemented to search for the optimal structures (i.e. the kind of neural networks, the number of inputs and hidden neurons) of neural networks which are used approximating a given nonlinear function. Two kinds of neural networks, i.e. the multilayer feedforward [1] and time delay neural networks (TDNN) [2] are involved in this paper. The synapse weights of each neural network in each generation are obtained by associated training algorithms. The simulation results of nonlinear function approximation are given out and some improvements in the future are outlined.

1   Introduction

We have some simulation results [3], [4], [5] on the function approximation application based on the neural network with fixed structures, in which, we tried one by one manually to search for the most suitable neural network structure for an multi-input one output nonlinear process in a temperature control system. Finding an alternative to automatically optimize the neural network architecture is our recent research emphasis.

To design the optimal architectures of artificial neural networks (ANNs), one attempt is implementing the evolutionary algorithms, e.g. evolutionary programming (EP) [6], genetic algorithm. There are also various designing methods relating to the automatically searching for the optimal architectures of ANNs, e.g. constructive and pruning algorithms [7].

In this paper, GA based algorithm is proposed to search for the optimal architecture of ANNs which are used to approximate a given nonlinear function [8], [9]. The chart flow of this method is illustrated by Fig.1.

2   ANN Structures and Training Algorithms

We mainly consider two kinds of ANNs in this paper, i.e. the general multilayer feedforward neural network and time delay neural network (TDNN). For the former one, the training algorithm is the improved back propagation algorithm. For the training algorithm of TDNN, please refers to [2]. To simplify the problem, we

make some limitations on TDNN structure, i.e. for any input node and hidden node in TDNN, there is only one step time delay.

> a. Randomly generate the first generation with population size N_p.
>
> *While (not terminated) do:*
> b. Decoding the individual of population (genotype) into the related neural networks (phenotype).
> c. Carry out the training process for neural networks decoded above and calculate the fitness for each phenotype.
> d. Carry out the GA operators upon the genotypes according to their fitness.
> e. Termination judgement.

Figure 1:   Chart flow of optimization algorithm

Figure 2 illustrates one TDNN with two input nodes, 3 hidden nodes and one output node.
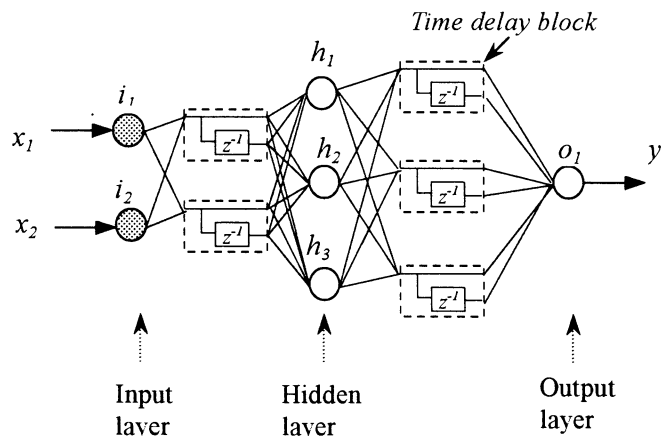
Figure 2: One example of TDNN

3 GA Based Optimal Algorithm

3.1 Fitness Function

The fitness of each fully trained network is calculated using Eq. 1.

$$f = \frac{1}{E}(1 + \alpha(1 - \frac{N_i}{N_{i\,max}}) + \beta(1 - \frac{N_h}{N_{h\,max}})) \qquad (1)$$

where f is the fitness of the neural network, E is the error of ANN, α and β are coefficients implying the influence of inputs number and hidden neuron number, N_{imax} and N_{hmax} is the maximum number of inputs and hidden neurons, respectively.

3.2 Encoding Algorithm

The first step before using GA operators is encoding the neural networks into binary strings called chromosome. A chromosome's characteristic is determined by the genes which are presented by binary bits in this paper. There are numerous encoding algorithms in neural network optimization. Because we emphasis our research on the neural network architecture optimization, the encoded strings just contain the architecture information of ANNs, i.e. the synapse weights of ANNs will not be involved in the encoded strings. The job of obtaining suitable weights is fulfilled by using the ANN training algorithms. The general description of encoding method is shown by Fig. 3.
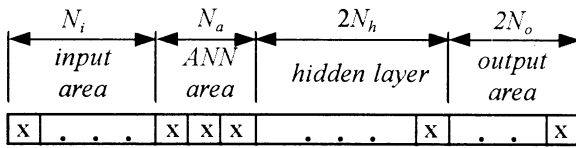
Figure 3: The chromosome presentation

Figure 3 shows one chromosome related to one neural network architecture, where x is one binary bit representing '0' or '1'. The chromosome consists of four parts, i.e. input area, ANN area, hidden layer area and output area, respectively. The length of each parts is N_i, N_a, $2N_h$ and $2N_o$ bits, respectively, where N_i, N_h and N_o is the number of inputs, hidden neurons and outputs of ANN, respectively.

In input area, one bit represents one input in input layer of neural network, with '1' representing the existence of the input and '0' representing the absence of the associated input. In the second portion of the string, three bits are assigned to represent different kinds of ANNs, with '001' representing multilayer feedforward neural net-works, '010' representing TDNN, etc. In hidden layer and output areas, two bits represent one neuron because we consider three kinds of active functions for each neuron, with '00' represents the absence of the neuron, '01'--- the neuron with linear function, '10' --- the neuron with sigmoid function, '11'

--- the neuron with hyperbolic tangent function, respectively.

Figure 4 shows one of chromosome generated by GA randomly, with the capacity of 4 inputs, 5 neurons in the first and second hidden layer, one neuron in output layer.
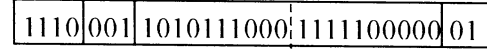
Figure 4: one of chromosome of neural network

3.3 GA Operators

Three GA operators, i.e. reproduction, crossover and mutation are implemented one generation by generation to search for the optimal architecture of neural network.

3.3.1 Reproduction

One commonly used technique is roulette wheel reproduction. It can be regarded as allocating pie-shaped slices on a roulette wheel to population members, with each slice proportional to the member's fitness. There are two drawbacks to this method: it is possible that some of the best individuals may not be reproduced at all, and thus their genes may be lost. Also, it is possible that the genetic operators alter the best chromosome's gene so that whatever was good about them is destroyed.

To improve the properties of roulette wheel reproduc-tion, an alternative called steady state reproduction is used in this paper. In this method, P_r percent of the best individuals will just be copied into the new generation, while the remained (100- P_r) percent individuals will carry out the crossover and mutation process. This method removes the drawbacks in roulette wheel, the best individuals will always reproduce because they are simply copied into the new generation and their genes are not changed by GA operators.

3.3.2 Crossover

Crossover is one of important operators in GA unlike the case in evolutionary programming (EP), in which crossover is not implemented and only mutation is carried out.

There are different possible crossovers in GA literature, e.g. one point crossover and two point crossover. For one point crossover, one part of the parent chromosomes are exchanged at the randomly selected point, another part is kept the same as before. The simplicity of one point crossover makes it be widely used in GA operators, but its drawback is obvious, that is the possible searching space is limited because only one point is chosen. In this paper, we use the two point crossover to over come the shortage of the

previous one.

3.3.3 Mutation

Mutations are important for keeping a bit of wildness and random search flavor to optimization process. In this paper, we use the general random mutation method, that means each gene in a chromosome changes its value (called allele) from '0' to '1', or from '1' to '0' at the given probability P_m, where P_m is set to 0.0025 in the following simulation.

4 Simulation

4.1 Training and Testing Data

Generally, the real data is divided into three groups for the training of neural network, i.e. training data, testing data and validation data. In this paper, we divided the real data into two groups, i.e. training and testing data. To consider the influence from both the training and testing data, a combined fitness is employed in estimating the feature of evolved neural network, see Eq.2.

$$f = \frac{\mu_1 f_{train} + \mu_2 f_{test}}{2} \qquad (2)$$

where f_{train}, f_{test} is the fitness of neural network based on the training data and testing data, respectively. μ_1 and μ_2 are the influence coefficients for training and testing data, respectively. In this paper, the coefficients are selected as $\mu_1 = 0.2$, $\mu_2 = 0.8$. It is reasonable for the fitness based on the testing data has more influence in the final estimation of neural network

4.2 The Approximation of Sine function

Firstly, the above algorithm is implemented to automatically generate a neural network which is used to approximated a sine function. The final generated neural network is one input node, 4 hidden neuron and one output time delay neural network (1-4-1). We define that there is one step time delay for each input and hidden neuron. The output the neural network compared with the real output is illustrated in Fig. 5. The mean square error (MSE) of this neural network via training steps is illustrated by Fig. 6.
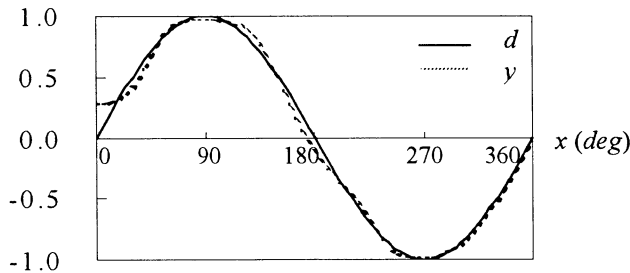
Figure 5: The network output for sine function approximation
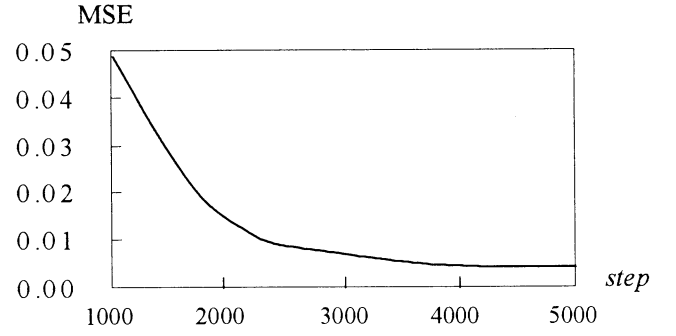
Figure 6: The network output MSE via training steps

After 5000 training steps, the network output MSE is 0.00354. Because the network MSE at beginning is 100 times more than the last one, it is difficult to illustrate the network whole MSE in one figure. Figure 6 shows the network MSE from 1000 to 5000 training steps.

4.3 The Approximation of a Nonlinear Function

The nonlinear function being approximated is multi-input one output one illustrated by Fig.7a - Fig. 7f. Where, x_1, x_2, x_3, x_4, x_5 are inputs, d is the output of the real system. There are 666 sets of input and output data as the real inputs and output sets. Two parts of these sets are separated, i.e. 333 training and 333 testing data.

Fig.7a

Fig. 7b

Fig.7c

Fig.7d

Fig.7e

Fig.7f
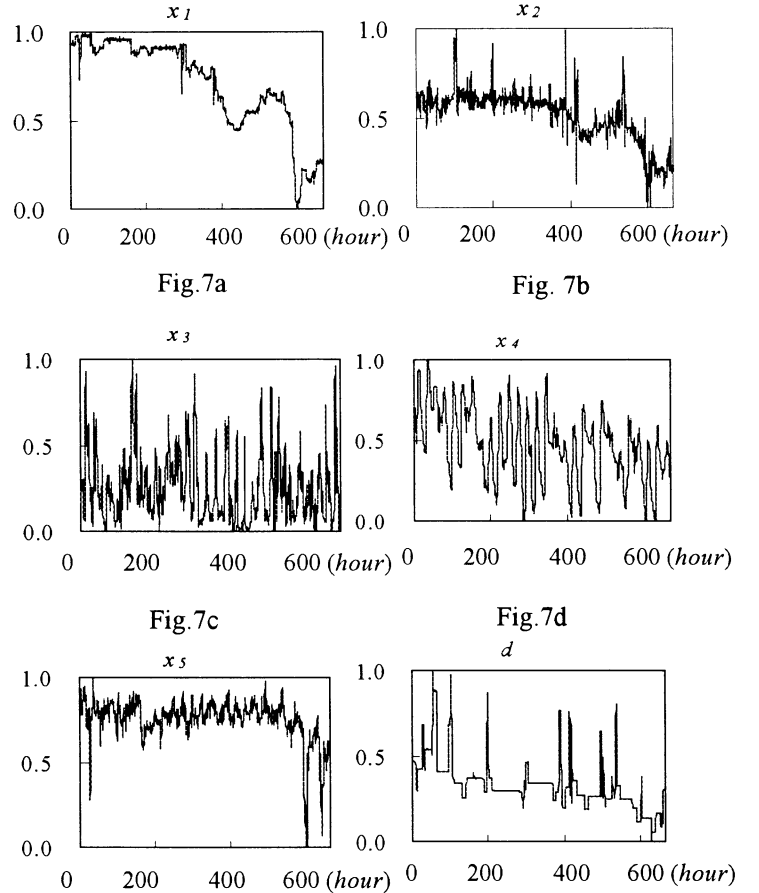
Figure 7: The Nonlinear Function

Each generated neural network by GA operator or randomly at start stage will be fully trained using the training data and then, its fitness will be calculated and the GA operators will be implemented to carry out the GA process. The system parameters are set as following. Population size N_p = 30, generation number N_g = 100, the input factor α = 0.025, the hidden neuron factor β = 0.1, reproduction factor P_r = 0.06, the mutation probability P_m = 0.002, using two point crossover method, the maximum number of hidden neurons N_{hmax} = 16, the maximum number of inputs N_{imax} = 5, the initial random weights are limited between ±0.3, the learning rate is 0.1, the momentum coefficient is 0.08. The final result are shown by Fig.8 and Fig. 9.
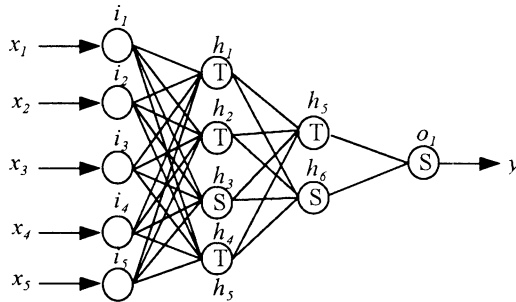
Figure 8: The best network generated by GA

Figure 8 is the best neural network evolved automatically by genetic algorithm with 5 inputs, 6 hidden neurons and one output. Figure 9 illustrates the neural network output y comparison with the system real output d. This network is evolved in 49 generation and is trained 8921 steps, with the training mean square error MSE_{train} = 0.0031, testing MSE_{test} = 0.0041, the fitness of this network f = 91.9.
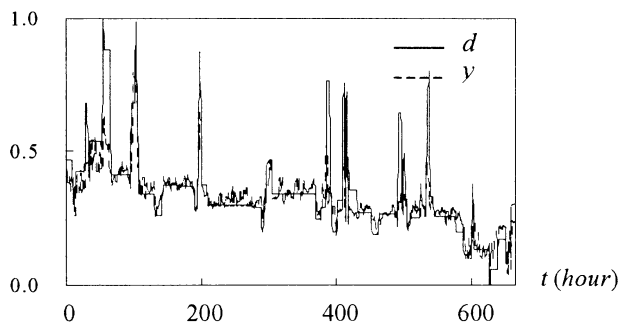
Figure 9: The network outputs compared with real outputs

5 Conclusion

We made some attempts to automatically search for the optimal architecture of neural networks employing genetic algorithm. The simulation results show that the presented method is suitable in nonlinear function approximation application. According to the simulation result, a feedforward neural network with 6 hidden neurons may get the mean square error 0.05. In the near future, we will add other kinds of neural networks and training algorithms to expand the searching ability of this method. Also, we are considering to use this method to other applications.

References

[1] Parisi R, Claudio EDD, Orlandi G and Rao BD (1996), A generalized learning paradigm exploring the structure of feed-forward neural networks, IEEE Transactions on Neural Networks 7(6): 1450-1459.

[2] Lin DT, Dayhoff JE and Ligomenides PA (1992), A learning algorithm for adaptive time-delays in a temporal neural network, Technical Report SRC-TR-92-59, Systems Research Center, University of Maryland.

[3] Sugisaka M and Liu ZJ (1998), The simulation of artificial neural network methods in cooler control using neurocomputer, Proceeding of the Third Workshop of Int. Institute for General Systems Studies, Qinhuangdao, China, July 26-28, 1998, pp.140-143.

[4] Sugisaka M and Liu ZJ (1998), The application of neurocomputer for controlling the temperature of cooler of sinter plant, Proceeding of XIII Int. Conference on System Science, Wroclaw, Poland, Sept. 15-18, 1998, Vol. III, pp. 352-358.

[5] Sugisaka M and Liu ZJ (1998), Artificial neural network and application in temperature control system, Proceeding of Korea Automatic Control Conference, Pusan, Korea, Oct. 15-17, 1998, pp. 260-264.

[6] Yao X and Liu Y (1997) A new evolutionary system for evolving artificial neural network, IEEE Transactions on Neural Networks 8(3): 694-713.

[7] Castellano G, Fanelli AM and Pelillo M (1997), An iterative pruning algorithm for feed-forward neural networks, IEEE Transactions on Neural Networks 8(3):519-531.

[8] Liu ZJ and Sugisaka M (1999), Neural network architecture optimization and application using genetic algorithm, Proc. of the 4th Int. Symp. on Artificial Life and Robotics, Beppu, Oita, Japan, Jan.19-22, 1999, Vol. 2, pp. 767-770.

[9] Liu ZJ and Sugisaka M (199), Optimization of neural network structures based on genetic algorithms, 11th SICE Symposium on decentralized autonomous systems, Nagoya, Japan, Jan. 18-19,1999, pp. 11-14.

The optimization of a neural network by genetic algorithm

Masanori Sugisaka ,○Minoru Itou
Department of Electrical and Electronic Engineering, Oita University,
Oita, 870-1192, Japan
E-mail:msugi@cc.oita-u.ac.jp   minoru@cc.oita-u.ac.jp

Abstract

The optimization of the structure and the learning of the neural network are necessary for making a neural network which has the ability for some purpose. The theory for generating an optimal neural network structure with the weight for a given purpose is necessary regardless of the processing for the purpose. The way of creating the better neural network from some neural network must be found as one of the ways.

Therefore, in this study, a neural network is optimized with Genetic Algorithm (GA). It is described that the optimization of neural network structure is performed about the problem of Exclusive OR(XOR). The algorithm uses uniform-crossover as the crossover method of the gene and changes mutation rate and the selection rate.

1.Introduction

The history of the study of GA is quite long. A lot of study is accomplished so far and results of a lot of studies are reported. GA modeled the mechanism of the evolution of the creature and the inheritance like engineering. However, in the conventional technique, it is known as the effective means to the various optimization problems in which the solution is difficult to find analytically.

Today, as for the computer, the ability to process improved leapingly. However, when doing advanced information processing, it isn't enough. Therefore, a way of the processing of the standing in the row based on the mechanism which does advanced parallel dispersion processing like the cerebral nervous circuit network is studied. It connects a lot of units which correspond to the nervous cell in the brain in these ways and it builds a circuit network and the study of the neural network which executes information processing by the pattern recognition and so on is done. The optimization of the structure and the learning of the neural network are necessary for making a neural network which has the ability for some purpose. The theory for generating an optimal neural network structure with the weight for a given purpose is necessary regardless of the processing for the purpose. The way of creating the better neural network from some neural network must be found as one of the ways. *[1][2][3]

2.The algorithm of simple GA
2-1.The setting of a virtual creature and environment

First, we fix the chromosome of the individual. We decide in what form GA tells the individual of the posterity what

content data from the individual of the parents in case of generation shift. A chromosome is generally composed of plural gene. The inner expression of the chromosome of each individual is called a genotype. We use the bit line which is 0 and 1 row as the symbol example which shows a genotype. It is each 0 or 1 bit that, in this case, is equivalent to the gene. Next, we fix the phenotype which is equivalent to the substance of each individual who has a genotype. Following the decision of the genotype, the phenotype, we fix a way of computing the fitness of each individual. The fitness is equivalent to the evaluation value about the point when assuming that the search point in the search space is each individual. It sets for the fitness of some individual not to be computed irrespective of the other individual and to be fixed from the relation with the other individual. The individual that the fitness is high is an excellent individual. The surviving probability or the probability of those individuals to make posterity makes become high. Oppositely, the individual that the fitness is low makes the individual who isn't adaptable to the environment successfully and is made to become extinct. This reflects the principle of the nature selection about the evolution theory. That is, the fitness is the measure which shows in which degree it is excellent, seeing from the surface which each individual calls surviving possibility. *[3]

2-2.The process of simple GA

The simple GA evolves a creature group by the procedure which is shown by Fig.1 after setting of the virtual creature and the environment which was described in 2-1. *[3]
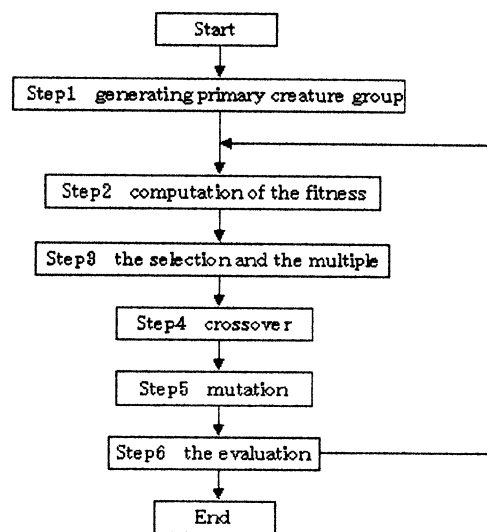
Fig.1 The process of simple GA

3. The optimization simulation of a neural network structure

The condition of the neural network to treat by this study is described.

(1)The unit which is equivalent to the nervous cell takes either of 0 (the non-exciting condition) or 1 (the exciting condition) as its own condition. This unit makes the following condition 1 when the amount of the input signal when it is bigger than 1 and in the case except it, it makes the following condition 0.

(2)The circuit network is the neural network which is composed of input unit group, hiding unit group, output unit group. The remaining number is put to the unit in order of the input unit, the hiding unit, the output unit. Supposes that there is not combination from the unit that the number is big to the small unit.

(3)The combination weight among the units is 1, -1 or 0 . 0 shows that there is not combination among them.

(4)First, make agree with the input signal of the problem to apply the value of the input unit after the value of each unit is made the early stages in 0 and the number makes a hiding unit, an

output unit transfer to the small unit in the condition in order. Then, we assume that it outputs the input signal about the value of the output unit after the condition transfer. But, here, the condition of each unit is supposed to be made the early stages of 0 every time it does signal input with once.

Input layer   Hidden layer   Output layer

——▶ weight 1   ——▶ weight– 1
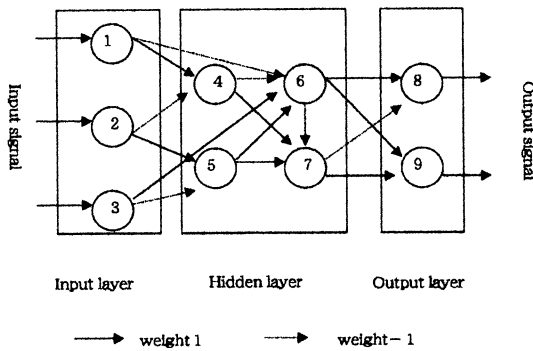
Fig.2   the example of the neural
network to treat in this sutady

		The remaining number of the unit								
		1	2	8	4	5	6	7	8	9
The remaining number of the unit	1	×	×	×	×	×	×	×	×	×
	2	×	×	×	×	×	×	×	×	×
	8	×	×	×	×	×	×	×	×	×
	4	1	-1	0	×	×	×	×	×	×
	5	0	1	-1	1	×	×	×	×	×
	6	-1	0	1	-1	1	×	×	×	×
	7	0	0	0	0	1	-1	×	×	×
	8	0	0	0	0	-1	1	-1	×	×
	9	0	0	0	0	0	1	1	×	×

Table 1 The table which shows the combination
Condition of each unit of Fig.2
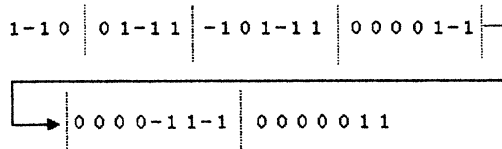
Fig.3   The genotype of the virtual

We assume that the neural network as shown in Fig.2 is a virtual creature and GA is applied for it. To define the genotype of the virtual creature, we use the table which shows the combination condition of each unit of 1 piece of neural network. Incidentally, the figure in the ◯ mark which shows each unit at the neural network in Fig.2 is the remaining number of the unit. Aa to first colomn in Table.1, each element shows the load of the combination from the unit which is to another unit. The value of each element, too, becomes one in these because a combination load is limited to either of 1, -1, 0. Also, the element of the × mark shows that it must not define the combination. Table.1 makes the letter line which arranged the element which is not a × mark in 1 in one line by the order in the lower right from the upper left as shown in Fig.3 the genotype of the virtual creature in this place.

$$f(I_i) = \frac{n_{correct}}{n_{total}}$$

Next, it defines fitness. Therefore, when signal input by the right answer of $n_{correct}$ is gotten after inputting a signal to the neural network which corresponds to some individual (I=1,2,...N) by n_{total}, it defines the fitness of the individual by the following equation.

It adopts the way of cutting a rate with the constant lower rank of the fitness as the selection method, and making up posterity with the crossover of the pair of the higher rank individual and keeping an individual total constant. As the way of the crossover, we use uniform-crossover. The uniform crossover is the crossover method which generates the genotype of the posterity by randomly choosing the gene which parents 1 or parents 2 correspond to as the gene of the posterity in the constant probability. In

this study, only when making the individual of the posterity with the individual of the parents, it sets for the uniform-crossover to occur always. In addition to the uniform crossover, we considers the mutation of the gene, too. *[1][2][3]

4.Conclusion

In this study, we fixed structure by the neural network, using GA. We used a uniform-crossover method as the crossover method of the gene, and we change a parameter variously and we adopted data for the application to the problem of XOR. Then, we confirmed the effectiveness when applying GA. In the future, we think that we want to apply the genetic algorithm which is a means having to do with an effect to the various optimizing problems and the search problems to the other field having to do with engineering, too.

Reference

[1]M.Sugisaka "The system theory and the control", Nikkankougyoushinbun-sha, in Japanese, pp.155-170, 1997
[2]S.Abe "Neural Network and Fuzzy system", Kindaikagakusha, in Japanese, pp1-7, pp60-65, 1995
[3]Y.Yonezawa, "Genetic algorithm -the information science of the evolution theory-", Morikitasyutupan, in Japanese, pp93-111, pp139-144, 1993

Search Optimum in multi-peak goal function space by

genetic algorithm

Masanori Sugisaka and Masahiro Shono
Dept. of Electric and Electronic Eng.
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   shouno@cc.oita-u.ac.jp

Abstract

Genetic algorithm is the comparatively simple basic principle that a creature imitated a proces of evolution. Resolution is known as current technique for difficult various optimization problem and search problems.

This paper investigates the problem of how to search optimul value through multi-peak goal function space by genetic algorithms. Main difficulties lay in that the investigated problem can not be described well by function and can not be analyzed numerically. To these problems, genetic algorithm is one of promising methods since the numerical model of problem is not needed

1.Introduction

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

2 .1.The search optimum problem

An optimizing problem is the problem to demand a maximum or the answer to minimize in the purpose function under the condition to have been given and generally, it is written as follows.

The purpose function $y(x) \rightarrow$ maximum

The agreement condition $x \in S$

Here, S shows an area with doing the can x of the variable can of which be taken and shows the set of the solution which meets an agreement condition.

The solution which belongs to area S is called a possible solution and sometime, the solution which maximizes purpose function $y(x)$ which is called the optimal solution.

When possible answer x is the real number, generally the inclination method which uses several cost function is used when continuing or it is possible to think that the purpose function is continuation like repartition and solving such a problem.

On the other hand, it is called a scattering optimizing problem in case of S' being a scattering set, e.g, in the case like the set of the integer and the nature number set.

For example, there are a patrol sales man problem and so on, combine and the much is formulated as the optimization problem by the scattering optimizing problem.

2.2. The scattering optimizing problem

As the approach to the scattering optimize problem, dynamic programming, branch-and-branch method, and so on, are considerd.

When it is difficult to demand the strict answer of the scattering optimizing problem (difficult in a lot of cases), if there is not a problem in case of practical use especially, it often substitutes by the approximate answer.

Also, the solution often can be efficiency well demanded if being an approximate answer. changes the real number variable into the binary letter line.

When thinking of the maximum value search problem in the numerous peak purpose function space as the example, continuation function space is changed into the scattering variable space by the scattering theory.

At this time, the set of the possible solution of the optimizing problem is changed into the scattering set and can be writeen as the scattering optimizing problem.

However, the scattering optimizing problem has the character of the combination problem basically whereas is making represent a continous variable by the scattering point in this case only.

At this point, it is an essential difference with the scattering optimizin problem.

2.3. Cording of the variable

In GA, we code the real number variable as the bit line with finite length in the area of the possible solution.

In the case of the space of the function of the purpose to have more than one real variable, more than one variable can have been codinged to one bit line.

As an example, when we code one variable x to the N bit 2 letter, line at x.min $\leq$ x $\leq$ x.max if giving an example, if showing the decimal number in order show by the optional binary letter line of the N bit in K, the decimal number which is shown by the binary letter line of the N bit is from 0 to 2^N -1.

$$x = x_{min} + (x_{max} - x_{min})K/(2^N - 1)$$

can do a change into the real number variable x from the binary letter line. Also, when codinging the real number variable x_i (x = 1,2,...,M) into the binary letter line with the $\sum_{i=0}^{M} N_i$ bit full length to have divided into N_i (I=1,2,..M) bits in $x_{i\,min} \leq x_i \leq x_{i\,max}$ ranges, the decimal number is shown by the optional binary letter line of the N_i bit in K_i.

At this time, the binary letter line is able for the real variable x to be changed into

$$x_i = x_{min} + (x_{max} - x_{min})K_i / (2^N - 1)$$

Here, coding to the binary letter line of the real number variable was described but the integer variable can be allocated into this letter line, too. The degree of the resolution is the index which shows in which degree the real number changes when only one decimal number changes.

Exactly, it is like the eyes of the network which catches a fish.

As the eyes of the network can catch the small fish that only the big fish is caught when rough but that it is small, it doesn't miss a change with the purpose functions which are few, too, when the degree of the resolution is high.

It is possible to think that this system is a necessary system to the optimal solution, too. For example, the decimal number which is shown by the binary letter line of 10 bits is from 0 to 2^{10}-1=1,023.
When only 1 decimal number changes, it changes into the range with the real number which is prescribed by the agreement condition by about 0.1 %.

3. The optimization technique

The characteristic to stand out and for GA is different from the random search is to say that GA searches in the numerous point standing in the row.

Therefore, it is possible to apply to not sticking only to the fact to merely look for the optimal point and searching various local answers, too.

Some tries which try to take a numerous peak with function were accomplished.

However, one tries to keep various-ness as a whole by the replacement of the individuals of either local. We use a concept such as sharing having to do with extremely the more effect because of the function optimiing than these operation.

This is the operation to attempt the relative uniformzation of the distribution of the function value by multiplying the weight which is small comparatively.

The function value in the part which a lot of points concentrate whereas multiplying big weight by the function value of the isolated point relatively.

This calls the sharing function, which satisfies

(1) $0 \le sh(d) \le 1$, $\forall d \in [0, \infty]$

(2) $sh(0) = 1$

(3) $\lim_{d \to \infty} sh(d) = 0$

by thinking of distance among coded letter line   sharing function.

Of course, However the various sharing functions can be considerd the following shaped sharing function is convenient.

$$Sh(d) \begin{cases} 1 - \left(d / \sigma_{share}\right)^{\alpha} & d < \sigma_{share} \\ 0 & otherwise \end{cases}$$

We correct a degree of the fitness as follows in case of making s_i individual function of the degree of the fitness $f(s_i)$ now, making an individual size N and distance's and sharing function's being chosen.

$$f(s_i) := \frac{f(s_i)}{m_i}$$

$$m_i = \sum_{j=1}^{N} sh(d(s_i, s_j))$$

Using sharing function characteristics, the following point is thought of.

(1) The prevention of the early stage convergence about the optimization problem

(2) The maintenance of the various-ness about the classification system

(3) The detection of the distribution of the local solution

4. Experiments and Results

Based on what was described above, we performed an experiments in two kinds of ways of experimenting.

(1) Two-point crossover(2X) : It is difficult the nearer two schema are on the inheritance child, the more they are destroyed. In other words, schema of the individual of the average equal to

or more than showing multiplies but intermingles 0 with the place where it should become 1.

(2) Uniform crossover(UX) : It is difficult to maintain schema by destroying nature of UX. In other words, 2X tends to preserve schema with function whereas UX go forward to the search of the various inheritance children while it destroys schema.

References

[1] Masatosh S.and Masahiro T(1995), "The Genetic Algorithm",An Asakura Cooperation pp,85-114.

[2] Rrohei Ishida((1997), "The basis and the application of Genetic Algorithm", The Morikita Cooperation pp,27-45.

[3] Hitoshi Iba(1994), " The basis of Genetic algorithm. It solves the mystery of Genetic algorithm.", The Orm Cooperationpp.pp,60-92.

[4] Hirano (1995)," The programming of Genetic algorithm",The personal media Cooperation pp.29-65

The optimal design of the fuzzy membership function

which used genetic algorithm

Masanori Sugisaka and Takaaki Kurimoto

Depatment of Electrical and Electronic Engineering

Oita university

1.Abstruct

In the field of the automatic control theory, it is roughly divided by "classical control theory" which is represented by the PID control and "advanced control theory" which is represented by the H_∞ control, the fuzzy control and so on. At present, the research of the control by advanced control theory are extensively done. In this research, we design fuzzy membership function using Genetic Algorithm.

Then we apply the design technique for a mobile robot with two independent driving wheels. In this research simulation study the effectiveness of this technique is explained.

2.Introduction

We consider of the termination control problem to move to the goal spot from a initial spot, where there is a two rings drive type movement robot as shown in Fig.1.

Here, D is the distance between the goal spot and the movement robot, the angle between the goal spot and the mobile robot is θ to accomplish, V are the velocity of the movement robot and ϕ is an azimuth of the moving of the mobile robot. $\Delta \theta = \theta (t) - \theta (t-1)$. $\Delta \theta$ is specified by $\Delta \theta$, θ, D and V become the input of the fuzzy control. Also, we use $5 \times 5 = 25$ control rule.

Here, We optimize the membership function of the fuzzy reasoning using genetic algorithm.
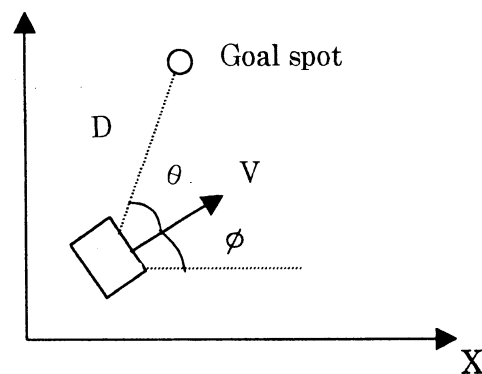
Fig.1 Mobile robot

3.Simulation

The purpose of the control action is for a mobile robot move from absolute coordinate (0,0) to the goal

spot given by (700,800). The fuzzy membership function is a triangle type as shown in Fig.2. We use the gene of the individual which have been coded by a central value of membership function with the gray code as shown in Fig.3. Central value ZE of membership function is 0. The ranges with central values of the variables are shown as follows V, D, θ, $\Delta\theta$ respectively.

$$-100 \leq V \leq 100$$
$$-1000 \leq D \leq 1000$$
$$-90 \leq \theta \leq 90$$
$$-90 \leq \Delta\theta \leq 90$$

In this study, We use a roulette selection. Also, we choose genes of two parents randomly and it is forming an individual in the following generation using uniform crossover. The probability of crossover is 0.5. Also, the mutation probability is 0.001. As for the evaluation of the individual, We make a reciprocal with total distance to the setting time fitness. Fitness is shown as follows.

$$fitness = 100000 / \sum_{t=0}^{30} |D(t)|$$

NB NS ZE PS PB

min   0   max
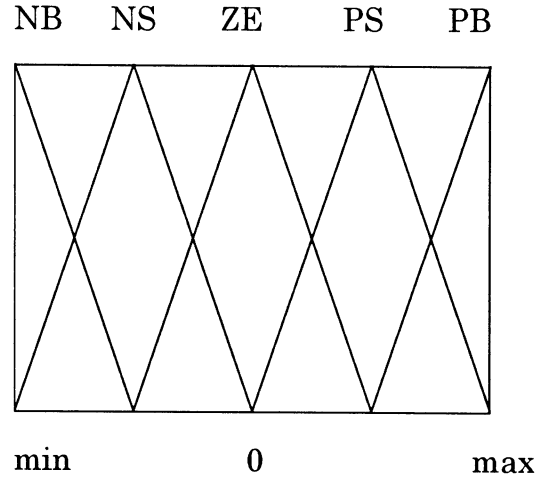
Fig.2 Fuzzy membership function

NB	NS	PS	PB	...

Fig.3 Genes of individual

4.Simulation result

The graph which shows the change of fitness is shown in Fig.4. The solid line is maximum fitness and the dotted line is average fitness. At the early generation, maximum fitness gets a high value. But average fitness is not convergent. Next, the graph which shows the velocity, the azimuth, the distance are shown in Fig.5, Fig.6, Fig.7. The solid line is the graph of the individual that has maximum fitness. The dotted line is the result of membership function which was established in the equal interval in the central value.

5.Conclusion

In this research, we applied genetic algorithm in the goal spot movement problem of the mobile robot. And, we showed the effectiveness of the algorithms proposed. To get a better result, it is necessary to improve an evaluation function. Also, because learning process is repeated until 50 generations in this time, the better result might be gotten when increasing the number of the generation shift.
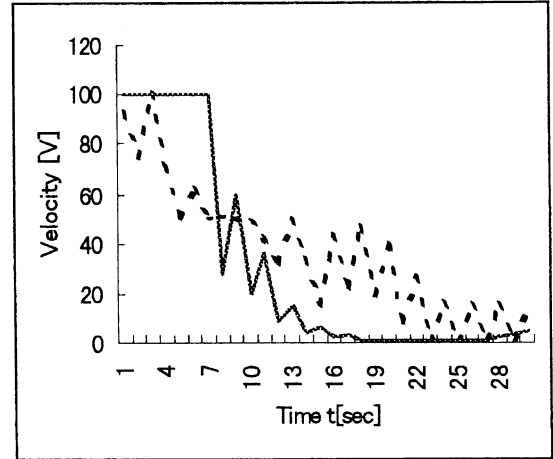
Fig.5 The velocity of mobile robot

Fig.4 The change of fitness
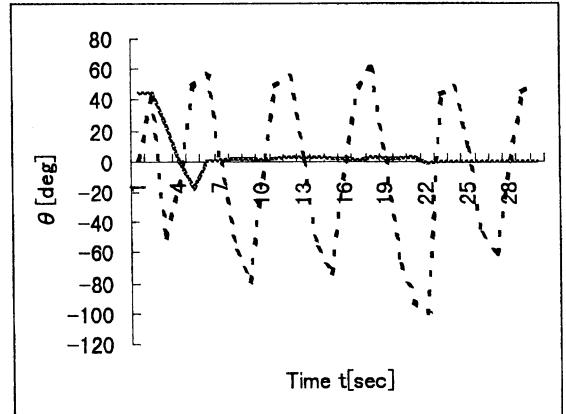
Fig.6 The azimuth of mobile robot
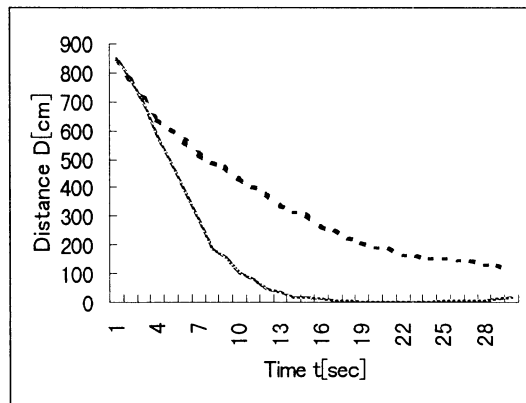
Fig.7 The distance of the mobile
robot

6.Reference

[1]Goldberg, D. E:Genetic Algorithm
in search, Optimization and
Machine Learning, Addision-
Wesley,1989

[2]R. A. Brooks: A Robust
LayeredControl Systems for a
Mobile Robot, *IEEE Robotics and
Automation*, Vol.2, No.1, pp. 14-
23,1986

[3]Holland, John H.:*Adaptation in
Natural and Artificial Systems.*
Ann Arbor, The University of
Michigan Press.

[4]Schaffer, J. David, Richard
A.Caruana,Larry J. Eshelman,
and Rajarshi Das:A study of
control parameters affecting
online performance of genetic
algorithms for function optima-
tion. In J.David Schaffer:
*Proceedings of the Third
International Conference on
Genetic Algorithms.* San Mateo,
Calif.: Morgan Kaufmann
Pablishers.1989

[5]Whitley, Darrell. GENITOR: a
different genetic algorithm. In
*Proceedins of the Rocky
Mountain Conference on
Artificial Intelligence.* Denver,
Colo.1989.

[6]Deb,Kalyanmoy, and David E.
Goldberg:An investigation of
niche and species formation in
genetic function ioptimization. In
J. David Schaffer, *Proceedings of
the Third International Co-
nference on Genetic Algorithms.*
San Mateo,Calif.: Morgan
Kaufmann Publisher-s.1989

[7]Goldberg, David E., and J.
Richardson. Genetic algorithms
with sharing for multimodal
function optimization. In John J.
Grefenstette, *Genetic Algorithms
and Their Applications :
Proceedings of the Second
International Conference on
Genetic Algorithms.* Hillsdale,
N.J.:Lawrence Erlbaum Asso-
ciates.

[8]Syaswerda, Gilberd: Uniform
crossover in genetic algorithms.
In J. David Schaffer, *Proceedings
of the Third International
Conference on Genetic Algo-
rithms.* San Mateo, Calif. Morgan
Kaufmann Publisher-s.1989

A new approach of self –organization
——from the DNA point of view

Yandong Zhang   Guangyi Cao   Xinjian Zhu

Fuel cell Institute, Department of automation

Shanghai Jiaotong University

ShangHai 20030, P.R.China

Abstract: This paper analyzes some basic biological facts of differentiation of cells and educes some rules that a logical-information model of differentiation must obey. Based on these analyses, we give a model to simulate the differentiation. The model partially explains how DNA works during the process of differentiation and at the same time the model gives a new approach of self-organize.

Keyword: Differentiation, Self-organization, logic-information model.

1. Preface

Since the 1950s the molecule genetics has developed greatly. But till now we still cannot answer an important question: how does a creature's structure emerge when it grows from an oospore to a complex and organized system? This process can be viewed as a self-organizing process: before the cells possessing the same DNA's differentiation, they are at the same level. But in the end the symmetry is broken and the cells possess very different proteins and play very different roles in the system and at the same time, structures are displayed. The mechanism of the differentiation of an embryo's cells when it growing is far too complex that is beyond the scope of this paper. In this paper, based on the analysis to some general accepted genetics facts, we build a logic-information model to simulate this self-organize process. The model partially explains how DNA works when it controls various cells to build various proteins and simultaneously keep the creature's structure stable from a logic-information (not chemical or biological) point of view. And at the same time the

model can be viewed as a mode of self-organize.

In chapter2 we will analyze the process of cells' differentiation from the logic-information point of view and educe several rules of a good logic-information model of DNA must obey. In chapter3 we will describe what such a modal may be and give one algorithm of such model. In chapter4 we will analyze and discuss the algorithm. In chapter5 we will discuss the implications and the perspectives of our work.

2. The logic-information analysis of cell's differentiation

To simplify the analysis, some assumption is made as followed: 1. We assume that all DNA molecules of the creature's cells are the same. 2. We assume that the DNA is the only matter in which the oospore stores its information of the embryo's structure. 3. We define all matters out of all the DNA molecules as the "Environment".4. We assume that cells were classified only according to their proteins, which means that two cells are considered in different classes if and only if their proteins are different. 5. We assume that there is a certain moment at which all cells of the creature belong to the same class.

All above assumptions are about the general rules that creatures must obey during the process of different-tiation. To facilitate the analysis, more assumption is introduced as followed regarding a specific creature's composition and structure.

We call this creature P. Then we assume that 6.the embryo of P has only two kinds of cells. One belongs to class A and the other belongs to class B. (Only cells belonging to class A possess proteins A, only cells belonging to class B possess proteins B and no cells possess both proteins A and proteins B). 7.the ratio of the quantity of cells belonging to class A and the quantity of cells belonging to class B (we call this ratio P's state in the following discussion) is a constant when differentiation is over.

Let's observe P's differentiation from stage T1 to stage T2. At stage T1 all cells of P are the same. At stage T2 cells of P belong to different classes and the ratio between quantity of cells belonging to class A and

quantity of cells belonging to class B becomes a constant. The total quantity of cells doesn't change from stage T1 to stage T2.

Then, from the logical-information point of view we can draw rules as follows:

Rule 1:The pieces of information imported from the Environment by DNA molecules are not all the same.

Because all cells' DNA molecules are the same, if the contrary proposition were true, different kinds of proteins would not be produced and differentiation would not take place.

A special example can be derived from this rule: not all DNA molecules importing nothing. That means, the Environment takes effect on some DNA molecules' determination about to produce which kind of proteins.

Rule 2:There are some channels through which the DNA molecules mutually exchange information about which kind of proteins they will produce or they have produced.

Before their decision to produce which kind of proteins, the DNA molecules must "know" P's present states. Otherwise, if every DNA molecules "freely" choose or decide to produce which kind of proteins, the structure features of embryo could not be stable.

However, the assumption2 implies that, out of the DNA molecules, something like an information center that monitors and records and controls the cells' differentiation cannot exist. Thus, only the above mentioned case is possible.

We can image that there are a set of "switches" on each DNA molecules and the state of such a "switch" on one DNA molecule would affect the states of "switches" on other DNA molecules. And the states of such switches' are corresponding to the changes of P's states. Only through some channels to exchange the information about these "switches" is it possible that the "switches" affect each other. That is to say, the information is dealt with in a distributed way.

Rule 3:For a DNA molecule, when decide to produce some kind of proteins is decided by the Environment or other DNA molecules or by both.

According to Rule 2, for a DNA molecule, to produce which kind of proteins is determined by P's present states. And the information of changes of the P's states is communicated through the information channels among DNA molecules. However we still do not know when the decision is made for a certain DNA molecule.

We call a cell that is prepared to produce proteins a "mature" cell. Before a cell matures, it doesn't export information to others. So when the first mature cell will emerge must be determined by the Environment.

But after the first mature cell emerges, other cells may grow in two possible ways. The first way is that in the environment there is a clock that awakes cells in turn. And the second way is that all cells (excepting the first mature cell) are activated directly or indirectly by the first mature cell. Furthermore, a hybrid way is possible either.

The above mentioned three rules are based on the simplified biological rules and a special kind of simplified creature. Still they fit for more general situations. They are the basic rules that a logic-information model of the differentiation must obey.

3. The logic-information model of the differentiation

We will use a program to simulate the DNA molecule in a cell and the computer where the program stored is used to simulate the cell that the DNA molecule belongs to. The computer with such a program is used as the basic unit of our model (just as a cell with a DNA molecule). To simulate how cells and DNA molecules work during the process of the differentiation and to simulate the self-organization, the program must achieve such functions: (1) While the same program is running at a number of computers, the results each computer producing are not all the same. (2) From the macro viewpoint, the results possess stable structure. (3) At which computer the program producing a certain result is not determinate. Just as in the last chapter, we assume that the embryo's structure is displayed by the ratio of different kinds of cells and we still use the assumptive creature P.

The algorithm of the program is as following:

Begin

 Constant: N_A,N_B;

 Stiff=0;Flag=1;n_A=0;n_B=0;

 Clock(self);

 If (flag=1)

 Begin

  x=Random(A,B);   Des=RanDes();

SendMessageOthers(x,Des); Produce(x);

  Stiff=1;

 End;

 While(Stiff=0)

 Begin

 End;

End;

MessageProcess(x,Des)

Begin

flag=0;

 If (x=A) n_A= n_A+1;

 If (x=B) n_B= n_B+1;

 If (Des=self) Begin

  Des=RanDes();Stiff=1;

  If ((n_A< N_A) and (n_B<N_B))Begin

  x=Random(A,B); SendMessageOthers(x,Des);

Produce(x);

  End Else Begin

   If (n_A< N_A) Begin

   x=A; SendMessageOthers(x,Des); Produce(x);

   End;

   If (n_B< N_B) Begin

   x=B; SendMessageOthers(x,Des); Produce(x);

   End;

  End;

 End;

End;

N_A, N_B —— the quantity of cells A and the quantity of cells B in the embryo.

Clock(self)——According to the *rule1* of chapter 2, we know that at least one of our basic unit of our model must import information from the environment. And this function is designed to achieve that object. In this algorithm, the function does nothing except that it is required to terminate at different moment when it is running in different computers. And the first computer where the function terminates corresponds to the first mature cell of the creature. In addition, in our algorithm it's required that there is only one computer where the function terminates firstly. The parameter *self* is used to identify different computers.

Random(A,B)—— the function that randomly returns A or B in the same probability.

SendMessageOthers(x,Des), MessageProcess(x,Des)—— According to the rules mentioned in chapter2, we know that the basic unit of our modal must exchange information mutually. The function *SendMessageOthers(x,Des)* is used to send messages to all the other computers and the function *MessageProcess(x,Des)* is executed after receiving the message sent by others. The parameters *x* and *Des* are transferred with the message sent by *SendMessageOthers(x,Des)*.

RanDes()——This function returns an address of the basic unit. The address is arbitrarily selected from the units that could import information. The basic units that have produced proteins would not be selected.

Self —— This parameter is used to identify different basic unit. It is different in different units.

Produce(x) —— the function output results. The value of *x* is *A* or *B*.

According to the *Rule 3* of chapter 2, we know that there are two basic modes to decide who is the next basic unit to produce proteins. In our algorithm, we use the second mode. The first mature cell is selected by "environment"(in our algorithm, it is decided by the function *Clock()*). And the next mature cell is selected by the parameter *Des* in the function *SendMessageOthers(x,Des)*. In the end all the other basic units are activated directly or indirectly by the first mature cell.

4. Some discussion about the algorithm

The algorithm shows what a logical-information modal of differentiation will be like in computers. But we must point out that the modal is not the only one to simulate the differentiation. We can view that point in three aspects: (1) The three rules we draw in chapter2 are the basic rules that all logical-information modal of differentiation must obey, but there are still many other possible ways to design a algorithm to obey these rules and to simulate the differentiation. (for example, we can change the mode to decide who is the next basic unit to produce proteins etc.) (2) The algorithm in chapter3 doesn't stipulate the details of all functions. We can implement these functions in very different ways. (3) To facilitate the description, the algorithm requires every computer to be connected mutually. In fact, only partially connected computers can achieve the same function.

The algorithm describes the phase in which the total quantity of cells doesn't change from stage T1 to stage T2. Actually the situation is very complex. Maybe we cannot extract such a phase from the process of differentiation of some kind of creature. However if we integrate more biological facts, we could design more delicate model to simulate the more complex process.

The algorithm simulates the differentiation of a very simple creature. The structure feature of the creature is displayed only by the ratio of quantities of its two kinds of cells. Actually creatures' structure is much more complex; And because we consider that one of the creature's cells only produce one kind of protein, we ignore that the effects that the protein exert to the DNA molecule. In fact, the process of differentiation is a process that DNA molecules and proteins mutually and continuously exchange information.

Despite of these blemishes, the algorithm describes the basic information transfer mode during the process of differentiation and provides a possible mode of self-organize. To deal with more complex situation, we need to design more complex model in the similar way as the algorithm describes.

5. Summary and conclusion

In this paper, we educe several logical-information rules about differentiation from some general accepted facts and design a model to display the process of differentiation in computers. This model partially explains that how such a self-organizing process takes place from the viewpoint of logical-information. At the same time, the model gives a mode of self-organization. According to this mode, the information is dealt with in a distributed way. And the information about the system's future is stored in the basic units at the beginning of the self-organizing process.

If we integrated more biological facts, we could design more delicate model to simulate the more complex process along the similar road of this paper. This kind of models maybe useful to biologists when they analyze the logical and the information transfer during the process of differentiation. And we can extract a new framework of self-organizing process from this kind of models. This framework can be used to explain some self-organization beyond the scope of the biology.

References

Kauffman S. A. (1993). The Origins of Order: Self-Organization and Selection in Evolution, Oxford University Press, New York.

Bak P. (1996). How Nature Works: The Science of Self-Organized Criticality, Springer, Berlin.

Scale Dependence of Fractal Dimensions in the Human Retinal Blood Vessel

Masahiko Monma, Yasuo Yonezawa [1]*and Takashi Igarashi [2]

[1] Complex Adaptive system Laboratory, Division of System Engineering,

Graduate school of Science and Engineering, Ibaraki University,

4-12-1, Nakanarusawa, Hitachi, 316-8511, Japan.

[2] Center for Medical Sciences, Ibaraki Prefectural University of Health Sciences

4669-2 Ami, Ami, Inashiki, Ibaraki 300-0394, Japan.

Abstract

In this study, we analyzed the complexity of normal human retinal vessels, in order to examine the correlation between form and function using fractal dimensionality. First, the box-counting method for determination of fractal dimensions was applied to the bifurcating structures of blood vessels distributed over the surface of the ocular fundus, considering the change of fractal dimensions in local area. These vascular patterns were demonstrated to be fractals characterized by distinct fractal dimension on different scales. Second, the box-counting method applied for image-processed patterns in order to examine the effect of thickness in patterns. The fractal dimension of blood vessels patterns vary with the scale. Third, Multifractal dimension was measured for the same retinal blood vessel. The multifractal dimensions depend on moment order q, and the dimendions gradually increase corresponding to the thickness of the blood vessels. In this paper, we present some results from the analysis of the fractal measurements.

1 Introduction

In previous research, it has been proposed that fractal structure is detected in several biological organs. Although a large number of researches have carried out into the application of fractal theory, little is known about their functionality of biological systems.

Namely, No speculation has take place concerning the generation of functions driven by fractal structures. Consider now the implication of the possess of fractal dimension to living organs. Most studies, however, have not focused that the influence fractal

---

*To whom correspondence should be address; E-mail yonezawa@life01.dse.ibaraki.ac.jp

structure had on the biological functions. For example, brain structure, blood vessel, protein surface, cell membranes surface, growth of bacterial colonies, dendrites of neuron etc are posses fractal structure.

These results are explains the properties of fractal and shows how they are now being used in biological structure. Many parts of each organs in living body have fractal structures described as above. The fractal dimension can be used to measure the differences between normal structures and those altered by several disease (abnormal structures).

The purpose of this research is to investigate how biological functions has been generated with fractal structures in living organisms. In order to elucidate the relationship with biological functions and fractal structures, we are proposed the analysis of eye blood vessels as typical complexity structure in biological organs.

An eye is one of the internal organs whose amount of blood is the most abundant in the internal organs which compose the living body. It is said that the amount of blood of every cellular tissue is about ten times of quantity of the cerebral gray matter when the circulation of the choroids and the retina are put together[1]. And, it is equal to observing a part of the brain blood vessel to observe a central retinal artery included in the retina, if it goes back to the eye artery and the internal carotid artery. As for the blood vessel of the retina, it knows that a change happens for the blood vessel wall and running due to the variety disease of the whole body clinically. In the past, Family et al, Matsuo et al, Master et al and Daxer calculated the fractal dimension of the retinal blood vessels[2'5].

Family et al insists that self-similarity exists in about one-digit scale range for dimensions measured from the mass dimension and the correlation function. Though there were differences in the method, fractal

dimension at that time was about 1.7. This dimension was possible by dealing with blood vessel growth with DLA (diffusion limited aggregation) model that it is controlled in the diffusion.

Matsuo et al insists that it is possible that the irregular form of the blood vessel bifurcation structure of a living body is quantified and classified. These vascular patterns were demonstrated to be fractals characterized by two distinct fractal dimensions on different scales.

Master et al couldn't distinguish a difference between the normality and the abnormality of the retinal blood vessels by using fractal dimensions. Because the method mentioned above were being researched with the premise that a living body has the same fractal structure, the knowledge which it referred to the change, which the component of the living body had scale dependence have not been get it [6].

We quantified complex structure of the retinal blood vessels by using fractal dimension of the limited part in this study. The aim of this study is to elucidate the role of fractals structure in complex adaptive system as living organisms. We employed the box-counting method to determine fractal dimensions for the bifurcating structures of blood vessels. Also we examined how the dilation or thinning in the image processing affect the partial fractal dimensions. Moreover we used the multifractal dimensions which extended the concept of the fractal dimension. Multifractal dimensions for moment order q were characterized by density of a pattern.

Our experiments suggested that the fractal dimensions of blood vessels patterns vary with the scale. These blood vessels patterns appeared to be characterized by two distinct fractal dimensions. Furthermore, we discussed a possible relationship of a fractal dimension D_L for larger scale range and a fractal dimension D_S for smaller scale range.

2 Materials and method

We determined the fractal dimensions of retinal blood vessel patterns (including both arteries and veins) using the box-counting method, and in order to determine the fractal dimensions in the part of the scale area, we calculated the partial fractal dimensions, which means slope of the part of the scale area. The partial fractal dimensions were calculated for the blood vessel patterns processed by the image processing of the dilation and thinning. Furthermore, multifractal dimensions were calculated for the same retinal blood vessels.

2.1 Fractal dimension

Although we can understand that the retinal blood vessels are distributed on a surface of a sphere in three dimensions, we assume that there are distributed in semi-two dimension in this study. In order to measure the fractal dimensions which generally expresses the self-similarity of a pattern, we used a definition based on covering the given structure with squares of various sizes (box-counting method).

The box counting method is based on finding the relation between the lattice constant, a and the number of squares, $N(a)$.

If the following formulas are realized between $N(a)$ and a at this time, it can be said that the fractal dimension D is then determined from the relation.

$$N(a) \propto a^{-\nu} \tag{1}$$

This index D is most important value by which is equivalent to the quantity which quantified the complexity of the pattern with self-similarity. The fractal dimension D was obtained from the slop of the log-log plots of $N(a)$ versus a.

$$D = ln\frac{N(a)}{ln(1/a)} \tag{2}$$

In addition, the partial fractal dimensions can be calculated from using the following formula

$$D_a = \lim_{a \to 0} ln\frac{N(a)}{ln(1/a)} \tag{3}$$

Furthermore, The multifractal dimensions, D_q generally are defined as a limit of the scale for the index number which is a moment order, q. The multifractal dimensions, D_q are then determined from the relation

$$D_q = \lim_{a \to 0} ln\frac{1}{q-1}\frac{lnZ(q)}{ln(a)} \tag{4}$$

Where
if q $\neq$ 1

$$lnZ(q) = ln\sum(p_i(d))^q \tag{5}$$

if q=1

$$lnZ(q) = (q-1)\sum p_i(d)lnp_i(d) \tag{6}$$

Where a probability $p_i(d)$ is the ratio of the number of
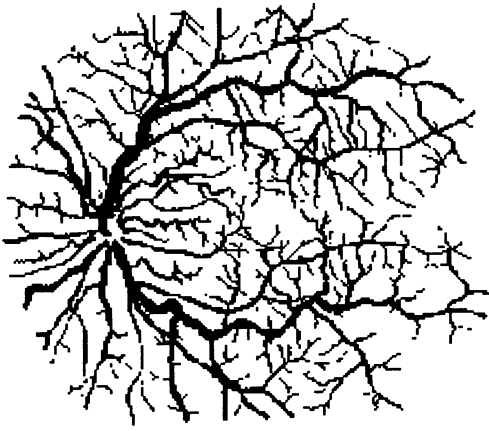
Figure 1: Schematic representation of retinal blood vessels

Figure 2: Relationship between a and $N(a)$

pixels within a square of size d in the lattice number i and the number of total pixels which composes a pattern. Multifractal dimension D_q is the capacity dimension for the case of $q = 0$ and the information dimension for the case of $q = 1$.

2.2 Method

With photographs, which extracted by tracing retinal blood vessels image from the fluorescent fundus oculi photograph of a normal man, we used a definition based on covering the given structure with squares of various sizes for the measurement of fractal dimension. The photographs were placed under a square lattice of lattice constant a, and the number of squares, $N(a)$, needed to cover the pattern were counted by computer-aided image analysis.Such measurements were made for different sizes of lattice constant a and we obtain a curve of $logN(a)$ plotted against $loga$. If the curve is almost linear with a slope D, then D can be interpreted as the fractal dimension. In addition, the partial fractal dimensions can be calculated from $logN(a_{i-1})$ and $logN(a_{i+1})$ and $loga_{i-1}$ and $loga_{i+1}$ (a_i is an arbitrary lattice constant).

3 Experimental result

An example of the images of the blood vessels in human retina is shown in Fig.1, which is a projection from the two-dimensional curved surface of the retina to the two-dimensional flat surface. The vascular patterns appearing in the photographs were limited in resolution; thus, vessels with a diameter of less than 15 m

could not be taken into consideration for the measurement. Fig.2 shows a typical curve of the log-log plots of $N(a)$ versus a measured for the vascular patterns. The graph is characterized by two subresions having different slops. For a shorter range, $N(a)$ decreases slowly with a small fractal dimension. But for a larger range, $N(a)$ decreases with a large fractal dimension. We observed the curve is characterized by two distinct fractal regions with a clear-cut cross -over point.

Fig.3 shows the plots of D_L for a larger range against D_S for a shorter range for each of 32 people inspected. From this figure, The fractal dimensions D_S for a shorter range were found to vary from 1.35 to 1.39, and the fractal dimension D_L for a larger range from 1.69 to 1.72 from person to person. Mutual relationship was not estimated from the results of measurements. When we plotted $(D_S - D_L)$ versus D_S, There seems to be a significant correlation between $(D_L - D_S)$ and D_S(Fig.4). We also define a partial fractal dimension D_a as a slop calculated in the immediate vicinity of the particular lattice constant. Fig.5 shows the curves of D_a plotted against lattice constant a for the effect of the image processing (the dilation or thinning). These results indicate that the partial fractal dimensions in the blood vessel pattern vary periodically. And the partial fractal dimensions D_a varies by image processing with box size 100 as a boundary. D_S for a shorter range increased gradually. But D_L for a larger range varies periodically. Fig.6 shows the curves of the relationship between the multi-
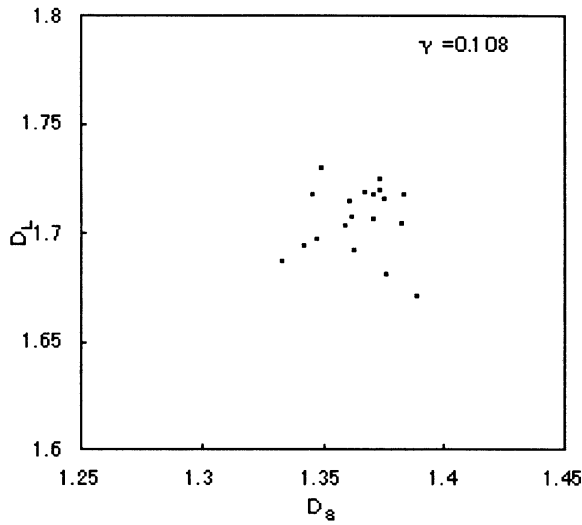
Figure 3: Relationship between D_S and D_L

Figure 4: Relationship between D_S and $D_L - D_S$ for the ratinal blood vessels

fractal dimension, D_q and a moment order, q. D_q vary according to the probability in a cell for each q. As compared with D_q in smaller size q, D_q in larger size q vary gradually according to image processing(dilation or thinning),which is equal to changes of diameter in blood vessels.

4 Discussion and Conclusion

We presented the results of an analysis on the blood vessels in human retina. We are obtained that vascular patterns in the retina are fractals characterized by two distinct subregions having different fractal dimensions. As the diameter of the blood vessels on a short scale range and the pattern of the blood vessels on a large scale range, the fractal dimensions change. The fractal dimensions are different in the border of the box size 100. And, the multifractal dimension indicate the differences in the density distribution of vascular pattern and changes greatly with diameters of blood vessels. Therefore, the blood vessel system is seemed to be composed of heterogeneous vessels.

These results might suggest that the blood vessels with a particular diameters form these own vascular patterns possibly for optimal configuration in the retinal tissue. Minimization of power losses at the blood vessels would contribute to the efficiency of the circulation. The geometry of branching would then be space filling and thus provide maximum area for the exchange of oxygen and metabolites. The significance

of a minimum volume optimality principle may be related to the fact that large and rapid change in flow rate are sometimes required in the circulation. This would be facilitated by a minimum volume circulation, where small changes in artery diameters would have a large effect on flow relative to pressure. Stephen et al suggested that the fractal structure which imitated the blood vessel encode arterial geometry and topology via function and facilitate fluid shear stress amplification in its peripheral vessels [7]. Thus, shear stress may be related to the vascular remodeling. However, further studies are needed to examine the effect of fluid shear stress and fluid pressure on epithelial cells.

These fractal branching models relate structure and function of the blood vessel and provide a mechanism to describe the spatially correlated distribution and the heterogeneity of blood flow. The present study indicates that competition among vessels should be extensively studied in detail by nonlinear dynamics in theoretical biology. In order to elucidate the function generated with multifractal structure in blood vessel, we are trying at the simulation of blood as fluids with multifractal pipeline.

5 Summary

Fractal analysis is useful for a branching pattern like retinal blood vessels. The fractal properties differ from those of the simple lines and curves. The
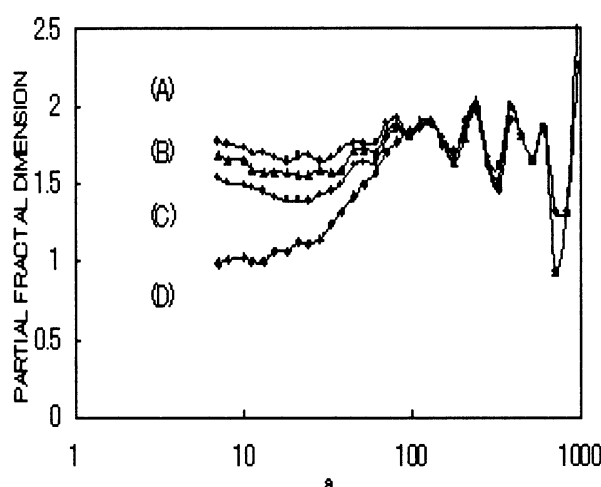
Figure 5: Relationship between fractal dimension and box size for the arterial tree of retinal blood vessel using image processing.
(A):dilation(25%).   (B):dilation(12.5%).   (C):original.
(D):thinning

Figure 6: Multifractal dimensions D_q of the ratinal blood vessels as a function of moment order q

issue of fractal dimension is surely not irrelevant to the issue of scales in biological structure (ex: blood vessel). A further important is that the transition of fractal dimension depend on observed scale are systematic. It follows from this that interactive analysis along these scopes would hope the elucidate of correlation between complexity function in blood vessels. The factors affecting the fractal dimension could be the shapes and diameter of retinal blood vessels. Two factors are considered to influence mutually. These results might suggest that the blood vessels with a particular diameters form these own vascular patterns possibly for optimal configuration in the retinal tissue. Consider now the implication of the process of fractal dimension to structure's of living organ's.Most studies, however,has not focused that the influence fractal structure had on the biological functions in several organs. In order to more clear the generation function from fractal structure, we are planning the model simulation for the solving of these results.

Acknowledgements

References

[1] "Physiology for the ophthalmology", Bunkodo, Tokyo, 1995.

[2] Family F., Masters B.R., Platt D.E., "Fractal pattern formation in retinal vessels", Physica D 38, pp. 98-103, 1989.

[3] Matsuo T., Takahashi M., Funata M., "Characterization of bifurcating structure of blood vessels fractal dimensions", FORM, Vol. 5, pp.19-27, 1990.

[4] Master B.R., "Fractal analysis of normal human retinal vessels", Fractal, Vol. 2, pp.103-110, 1994.

[5] Daxer A.,"Mechanisms in retinal vasculogenesis:an analysis of the spatial branching site correlation", Curr-Eye-Res, Vol. 14, No. 4, pp.251-254, 1995.

[6] Mandelbrot (Hironaka supervision of translation), "The Fractal Geometry of Nature", Nikkei science, Tokyo, 1995.

[7] Stephen H.Bennett, Boyd W.Goetzman, Jay M.Milstein, Jatinder S.Pannu, "Role of arterial design on pulse wave reflection in a fractal pulmonary network", J. Appl. Physiol., Vol. 80, No. 3, pp.1033-1056, 1996.

[8] Monma M., Yonezawa Y. and Igarashi T., "Analysis of correlation between complexity and function on the Human Retinal Blood Vessels", *Proc Advanced workshop of complex Functionary systems (in Japanease)*, pp.9-15, 1998.

[9] Larry S.Liebovitch, "Fractal and Chaos: Simplified for the lifesciences", *Oxford University Press*, 1998.

[10] Yasuo Yonezawa, "Introduction of the Emergent Functional Systems" and "Multiple Descriptional System Theory for Emergent Function" *in Proc of the 1st workshop on sci-ence for Emergent functional Systems(in Japanese)*, pp.12-20 and pp.89-92, 1998.

Mathematical Analysis on An Immune System

Masahiro Kawakita, Seiichi Shin
Dept. of Mathematical Engineering and Information Physics,
Graduate School of Engineering,
The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656 Japan

Abstract

This paper deals with mathematical analysis based on the immune model formulated by Shirai et al[1]. The model considered in this paper is not a biological one, but an engineering system based on immunology.

There are a lot of studies that deal with engineering application of immunology. These are, however, concerned mainly with application problems, not with theoretical analyses.

In this paper, the immune system is analyzed in the mathematical senses. Nonlinear simultaneous differential equations are provided through improving the model formulated by Shirai et al. Based on these equations, system behavior is obtained in which neighborhood of the equilibrium points in 2-antibodies case. System behavior in neighborhood of the equilibrium points is obtained by calculating eigenvalues of a coefficient matrix of linearized equations. The reserch of Shirai et al. is reconsidered form this view point.

1 Introduction

Recently, there are a lot of studies that deal with engineering application of immunology, for example, associative memory, failure detection, and so on ([1], [2], [3]). In these studies, information processing mechanism of immune network is applied to engineering system.

Immune network is a hypothesis suggested by Jerne[4]. This implies as follows: A class of antibodies suppresses a class of antibodies. Same as this, the class of antibodies suppresses another class of antibodies. This causes network of interaction between antibodies, and the construction induces information processing mechanism of immune system.

By the way, these engineering applications are, however, concerned mainly with application problems, not with theoretical analyses. Thus, in these studies, immune system is considered as system that may have guarantied performance. Immune system is thus applied to problem which has structure resembles to it, with expectation of show same performance. For example, in [1], immune model is formulated suitable for computer simulation, and applied it to select robot action. In the paper, robots and its actions are corresponding to antibodies and antigens respectively, and robot is selected by a numerical simulation in a computer.

Then this paper analyzes immune system in the mathematical sense. This paper derives a model suitable for mathematical analysis by improving to model formulated by Shirai et al [1]. The main part of the improvement is relaxation of coefficients. Coefficients of the original model take discrete values, because the model is formulated to computer simulation. Then, this paper relaxes the coefficients to continuous values suitable for mathematical analysis.

Based on these equations, system behavior is obtained in which neighborhood of the equilibrium points in 2-antibodies case by calculating eigenvalues of coefficient matrix of linearized equations. The reserch of Shirai et al. is reconsidered form this view point.

2 Relaxed immune model

By improving to the model formulated by Shirai et al [1], a model suitable for mathematical analysis is obtained as follows:

$$\frac{d}{dt}x_i(t) = x_i(t)^2(1 - x_i(t))$$
$$\times \left(\frac{\sum_{j=1}^{N} m_{ji}x_j(t)}{N-1} + m_i - k_i \right) \quad (1)$$

$x_i(t)$: acivity magnitude of antibody i ($0 \leq x_i \leq 1$)

m_{ji} : interaction magnitude from j to i
$\quad (-1 \leq m_{ji} \leq 1, m_{ii} = 0)$

m_i : activity magnitude of antigen $(0 \leq m_i \leq 1)$

k_i : death rate of activity of i $(0 \leq k_i < 1)$

This equation involves Shirai model as a special case.

3 Analysis by linearization

This section deals with analysis on 2-antibodies system, which is most simple application of the model.

3.1 2-antibodies model

2-antibodies system of the model is written as follows: $N = 2$ in (1),

$$\dot{x_1} = x_1^2(1 - x_1)(mx_2 - k_1 + m_1) \qquad (2)$$
$$\dot{x_2} = x_2^2(1 - x_2)(nx_1 - k_2 + m_2) \qquad (3)$$

x_1, x_2 : activity magnitude of antibody 1,2

m : interaction magnitude from antibody 2 to 1

n : interaction magnitude from antibody 1 to 2

k_1, k_2 : death rate of activity of antibody 1,2

m_1, m_2 : activity magnitude of antigen 1,2

Now, with $I \equiv -k_1 + m_1$, $J \equiv -k_2 + m_2$, above equations written as

$$\dot{x_1} = x_1^2(1 - x_1)(mx_2 - I) \qquad (4)$$
$$\dot{x_2} = x_2^2(1 - x_2)(nx_1 - J) \qquad (5)$$

3.2 Equilibrium points

Equilibrium points of equation (4),(5) is points which satisfy $\dot{x_1} = 0, \dot{x_2} = 0$. Such points are $(x_1, x_2) = (0,0), (1,1), (1,0), (1,1), (\frac{J}{n}, \frac{I}{m})$. In these points, former four are derived by sigmoid, and last one is essential equilibrium which worth to be analyzed.

3.3 Linearization

(4),(5) are nonlinear differential equations, so it is difficult to analyze behavior of the equation. Then, analysis is performed on behavior of linearized equations in neighborhood of equilibrium point.

We use the following linear transformation in order to shift equiblium point $(\frac{J}{n}, \frac{I}{m})$ to $(0,0)$:

$$X = x_1 - \frac{J}{n} \qquad (6)$$

$$Y = x_2 - \frac{I}{m} \qquad (7)$$

And to take linear terms, (4),(5) yield to

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = D \begin{pmatrix} X \\ Y \end{pmatrix} \qquad (8)$$

where

$$D = \begin{pmatrix} 0 & mJ^2(n - J)\, n^3 \\ \frac{nI^2(m-I)}{m^3} & 0 \end{pmatrix} \qquad (9)$$

This is the linearized equation of nonlinear differential equations in the neighborhood of equilibrium point.

A vector field is locally equivalent to its linear part at a hyperbolic equilibrium point (Hartman and Grobman's theorem [5]). Eigenvalues of coefficient matrix D are

$$\lambda = \pm \frac{IJ}{mn} \sqrt{(m - I)(n - J)} \qquad (10)$$

Behavior of equation hence depends on sign of $(m - I), (n - J)$.

3.4 The case of equilibrium point exists in the domain

By $0 \leq x_1, x_2 \leq 1$, Coefficients satisfy

$$(0 \leq I \leq m \text{ or } m \leq I \leq 0) \qquad (11)$$

and

$$(0 \leq J \leq n \text{ or } n \leq J \leq 0) \qquad (12)$$

when equilibrium point is in domain of (4),(5).

This means

$$0 < \frac{activity of antigen}{death rate + interaction magnitude} < 1 \qquad (13)$$

Then, system behavior is analyzes in next 4 cases.

1. Case a : $0 < I < m, 0 < J < n$ Eigenvalues of D

$$\lambda = \pm \frac{IJ}{mn} \sqrt{(m - I)(n - J)} \qquad (14)$$

take real values.

Hence, behavior of (4),(5) is locally equivalent to (8) in neighborhood of equilibrium point.

With

$$P = \frac{1}{2} \begin{pmatrix} \sqrt{\frac{n^3}{mJ^2(n-J)}} & -\sqrt{\frac{m^3}{nI^2(m-I)}} \\ \sqrt{\frac{n^3}{mJ^2(n-J)}} & \sqrt{\frac{m^3}{nI^2(m-I)}} \end{pmatrix} \qquad (15)$$

(8) is
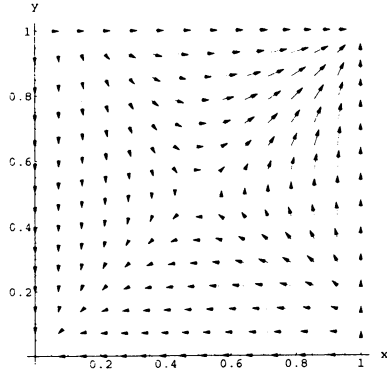
Figure 1: Vector field in case a

Figure 2: Trajectory of linear part in case c

$$\begin{pmatrix} \dot{X}' \\ \dot{Y}' \end{pmatrix} = P \begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = PDP^{-1} \begin{pmatrix} X' \\ Y' \end{pmatrix}$$

$$= \pm \frac{IJ}{mn} \sqrt{(m-I)(n-J)} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X' \\ Y' \end{pmatrix} \quad (16)$$

Then the equilibrium point is a saddle point (Figure 1)).

In neighborhood of $(x_1, x_2) = (\frac{J}{n}, \frac{I}{m})$, if initial value (x_{1_0}, x_{2_0}) satisfies

$$x_{2_0} < \frac{I}{n} - \frac{n^2 I}{m^2 J} \sqrt{\frac{m-I}{n-J}}(x_{1_0} - \frac{J}{n}) \quad (17)$$

(x_1, x_2) converges to $(0, 0)$, if initial value (x_{1_0}, x_{2_0}) satisfies

$$x_{2_0} > \frac{I}{n} - \frac{n^2 I}{m^2 J} \sqrt{\frac{m-I}{n-J}}(x_{1_0} - \frac{J}{n}) \quad (18)$$

(x_1, x_2) converges to $(1, 1)$.

2. Case b : $m < I < 0, n < J < 0$

$$\lambda = \pm \frac{IJ}{mn} \sqrt{(I-m)(J-n)} \cdot i \cdot i$$

$$= \mp \frac{IJ}{mn} \sqrt{(I-m)(J-n)} \quad (19)$$

Eigenvalues of D take real values, then behavior of (4),(5) is locally equivalent to (8) in neighborhood of equilibrium point.

Then, the equilibrium is a saddle point.

3. Case c : $0 < I < m, n < J < 0$

$$\lambda = \pm i \frac{IJ}{mn} \sqrt{(m-I)(J-n)} \quad (20)$$

The eigenvalues take imaginary values, hence equilibrium of D is not hyperbolic.

Then, behavior of (4),(5) is cannot be known by (8) precisely.

Here analyzes behavior only on linear term.

With

$$P = \begin{pmatrix} -\frac{n^2}{J\sqrt{J-n}} & 0 \\ 0 & \frac{m^2}{I\sqrt{m-I}} \end{pmatrix} \quad (21)$$

(8) yields

$$\begin{pmatrix} \dot{X}' \\ \dot{Y}' \end{pmatrix} = \frac{IJ}{mn} \sqrt{(m-I)(J-n)}$$

$$\times \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} X' \\ Y' \end{pmatrix} \quad (22)$$

Thus, the equilibrium is a central point of clockwise rotation.

Figure 2 shows a trajectory of (8) in $m = 1, n = -1, I = 0.5, J = -0.5$, initial value $(x_{1_0}, x_{2_0}) = (0.5001, 0.5001)$.

4. Case d : $m < I < 0, 0 < J < n$

$$\lambda = \pm i \frac{IJ}{mn} \sqrt{(I-m)(n-J)} \quad (23)$$

The eigenvalues of D take imaginary values, then the equilibrium is center point of counterclockwise rotation.

$-240-$

Table 1: System convergence point in $m > 0, n < 0$

		(A)	(B)	(C)
		n<0<J	n<J<0	J<n<0
(1)	0<m<I	(0,0)	(0,1)	(0,1)
(2)	0<I<m	(0,0)		(1,1)
(3)	I<0<m	(1,0)	(1,0)	(1,1)

3.5 Discussion

When condition (13) is saticefied, system convergence to stability is guarantied with linearized equations in following cases:

- in case a, (x_1, x_2) converges to $(0,0)$ or $(1,1)$

- in case b, (x_1, x_2) converges to $(0,1)$ or $(1,0)$

By the way, in Shirai model, it is assumed that antisymmetry of interaction, $m_{ji} = -m_{ij}$. This assumption is satisfied in case c or case d. In this condition, it is difficult to know precise behavior of system in neighborhood of equilibrium by linearized equation. But dominant behavior is oscillation, so system evaluation by threshold is not a reasonable. Thus, under condition (13), 2-antibodies Shirai model does not have desired behavior.

The next subsection then analyzes in case of condition (13) is not saticefied.

3.6 The case of equilibrium point exists out of the domain

If system doesn't satisfy (11) and/or (12), then equilibrium point is not in domain of (4),(5), hence (13)is not saticefied. In this case, system converges to one of equilibrium points which derived by sigmoidal function.

Table 3.6 shows convergence point in case of $m > 0, n < 0$.

The blank column of (B)-(2) in table 3.6 is corresponding to case c. For example, if the system state transits from (B)-(2) to (C)-(1), condition (13) is broken as on (m, I) . Then

$$\dot{x_1} = x_1^2(1 - x_1)(mx_2 - I) < 0 \qquad (24)$$
$$\dot{x_2} = x_2^2(1 - x_2)(nx_1 - J) > 0 \qquad (25)$$

are satisfied thus (x_1, x_2) converges to $(0, 1)$.

By shifting of equilibrium to out of range, system state transit from oscillation to stability.

4 Summary

In this paper, equations which are suitable for mathematical analyses are derived from model of immune system by Shirai et al. Based on the equations, system is analyzed in 2-antibodies case. Then, the work of Shirai et al. is reconsidered form this view point.

This paper analyzes of 2-antibodies system, which is most simple case, but it is necessary to study about a system consists of 3-antibodies or more. This model considers activity of antigen as constant. But it is required that treat it as variable to dynamical system analysis in case as of subsection 3.6.

References

[1] Y.Shirai, Y.Watanabe, A.Ishiguro, Y.Uchikawa : Emergent Evolution of Behaviour Arbitration System Based on the Immune system, Trans. Institute of Electrical Engineers of Japan, Vol. 118-C, No. 10, pp. 1465-1471 (1998)

[2] Y.Ishida, F.Mizessyn : Sensor Diagnosis of Process Plant by an Imune-Based Model, *Trans. Institute of Systems, Control, and Information Engineers*, vol. 7, No. 1, pp. 1-8 (1994)

[3] C.J.Gibert, T.W.Routen : Associative Memory in an Immune-BasedSystem, Proc. National-Conference-on-Artificial-Intelligence.v2, pp. 852-857 (1994)

[4] N.K.Jerne : Towards the network theory of the immune system, Ann. Immunol. (Inst.Pasteur), 125C, pp.373-378 (1974)

[5] J.Palis and W.deMelo : Geometric Theory of Dynamical System:An Introduction, Springer-Verlag (1982)

Adaptive Iteration Learning Control of Upper Limb Motion with Functional Neuromuscular Stimulation(FNS)

Huaiyu Wu,   Zhaoying Zhou,   Shenshu Xiong,   Ganghua Zhang
Department of Precision Instruments & Mechanology
Tsinghua University
Beijing 100084, P. R. China
E-mail: huaiyu_wu@263.net

Abstract

This paper introduces a new optimum closed-loop control strategy for upper limb motion, i.e. the adaptive iteration learning control. Firstly, the FNS control system and its basic principles are presented. Then, according to the nonlinear phenomena of the limb muscles, the discrete time-varying nonlinear model with undetermined and unexpected perturbations is defined, and a general expression based on the adaptive iteration learning control algorithm is developed, and the control algorithm structure block diagram is presented. Finally, based on multi-purpose FNS limb motion control system, the clinical experiments on motion trajectory-following of elbow flexion and wrist flexion were conducted by means of the adaptive iteration learning control algorithm and conventional control algorithm. The results of clinical studies have demonstrated that the adaptive iteration learning control algorithm is more suitable for the improvement of the dynamic response characteristics and the stabilization of limb motion than conventional control algorithm. Furthermore, the stimulated patients have not any bad physiological reactions because the output electrical stimulation pulses generated by the adaptive iteration learning control algorithm vary gently.

Key words: *FNS; iterative learning control; limb motion; feedback control*

1. Introduction

Functional neuromuscular stimulation (FNS) is one of new approaches in modern rehabilitation engineering. One area of intense current interest is the restoration of functional movement to the extremities paralyzed people. A major impediment to the development of satisfactory control systems for the functional neuromuscular stimulation of muscle has been the highly nonlinear, time-varying properties of electrically activated skeletal muscle which make control very difficult to achieve and limit the utility of open-loop FNS control systems. Therefore, feedback control can be used to increase the linearity of input-output behavior, and to improvement repeatability by decreasing system sensitivity to parameter variations (and other internal disturbances) as well as decreasing sensitivity to load changes (and other external disturbances). The use of closed-loop controllers for regulation of electrically stimulated muscle contractions has been investigated by many researchers. Applications of these feedback controllers in neuroprosthetic devices for restoration of extremity function has been limited, however, by technical problems associated with the development and mounting of adequate output sensors as well as by lack of robust controllers and stable methods of excitation.

The iterative learning control method is receiving increasing attention as an alternative for controlling uncertain dynamic systems in a simple manner, which is proposed by Uchiyama[1] and elaborated as a more formal theory by Arimoto[2] and some other researchers[3,4,5]. The ideal is based on the use of repeated trials of tracking of a preassigned trajectory. During each trial, the current trajectory difference between the real trajectory and the reference trajectory are recorded to be exploited by a suitable algorithm in the next trial, with the aim of progressively reducing the trajectory errors. It is a recursive online control technique that relies on less calculation and requires less a priori knowledge about the system dynamics because the algorithm is independent of the plant dynamics.

In this paper, we present the FNS system based on the iterative learning control scheme and then analyze the iterative learning control algorithm. In comparison with the other method, based on multi-purpose FNS limb motion control system, the clinical experiments on motion trajectory-following of elbow flexion and wrist

flexion were conducted by means of the adaptive iteration learning control algorithm and conventional control algorithm. The results of clinical studies have demonstrated that the adaptive iteration learning control algorithm is more suitable for the improvement of the dynamic response characteristics and the stabilization of limb motion than conventional control algorithm.

2. FNS System

2.1 FNS System

A typical FNS system comprises several units such as reference inputs, computer, stimulating map, multi-channel stimulators, safeguard devices, angle sensors, signal transformer and so on, as shown in Fig. 1
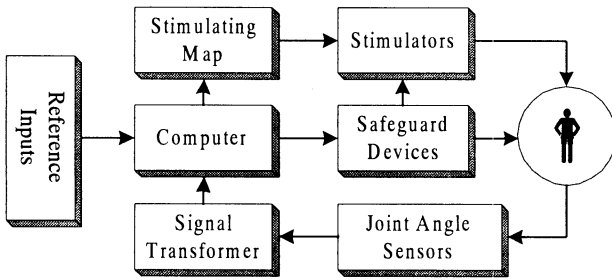
Fig. 1   FNS system block diagram

2.1 Mechanism

According to the specified motion control patterns or desirable motion control strategies, the electrical pulse signals (their amplitude, frequency and width are controllable) are generated by means of the computer associated with a special control algorithm. Then, the pulses are mapped, amplified and transformed into required stimulating sequences that are directly put on the joint-controllable-muscles (e.g. biceps brachii, long palmar muscle and et al.) via the stimulators. Consequently, the musculoskeletal system is generated a desired posture or movement because the movement neural is stimulated. During the movement nerves are stimulated, the advanced nerve center are also actuated though the spinal cord by the afferent neural. It is helpful to cerebral cortex to be excited after repeated movement pattern signals are applied. The basic functions of paralytic limbs can be improved permanently, and finally the movement functions can controlled or partly recovered by the patient.

In closed-loop FNS system, the limb movement states are measured by means of the angle sensors mounted on the motion axis centers of the elbow joint and the wrist joint. Those state signals are transformed to FNS computer after having been preprocessed by the signal transformer.

2.3   Measurement Scheme of the Joint Angles

The task of the control system was to specify a set of muscle stimulation parameters that would generate a desired movement of the skeletal system. In this paper, the musculoskeletal system (elbow and wrist) to be controlled consists of two muscles (biceps brachii and long palmar muscle). The performance of the FNS system based on a certain control algorithm is primarily described by the angle-trajectory-following curves of both the elbow motion and the wrist. The measurement scheme of both elbow joint angle and wrist joint angle is shown in Fig. 2
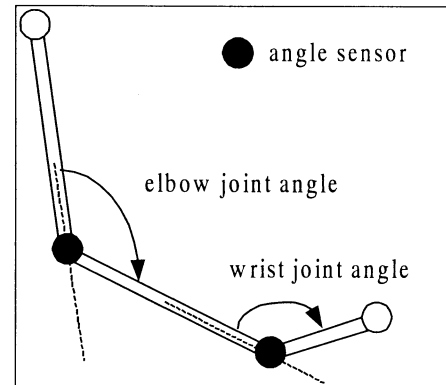
Fig. 2   Measurement scheme of the two joint angles

3.   Iterative Learning Control Algorithm

Consider the following discrete time-varying nonlinear system associated with undetermined and unexpected perturbations in FNS limb motions.

$$\begin{cases} x^i(k+1) = f(x^i(k),k) + B(x^i(k),k)u^i(k) + \omega^i(k) \\ y^i(k) = C(k)x^i(k) + v^i(k) \end{cases}$$

$$(1)$$

where $x^i(k) \in R^n$, $u^i(k) \in R^m$, $y^i(k) \in R^r$ are the state vector, input vector and output vector; i is the ith iteration; k is the discrete time , $k \in [0,N]$;

$C(k) \in R^{r \times n}$ is   the   output   matrix ; function

$f(\cdot,\cdot):R^n \times [0,N] \mapsto R^n$ and

$B(\cdot,\cdot): R^n \times [0,N] \mapsto R^{n\times m}$ meet the Lipschitz conditions associated with the state vector $x^i(k)$, i.e.

$$
\begin{cases}
\| \Delta f^i(k) \| \le k_f \| \Delta x^i(k) \|, \\
\| \Delta B^i(k) \| \le k_B \| \Delta x^i(k) \|, \\
\forall i, \forall k \in [0,N], \exists k_f > 0, k_B > 0, s.t.
\end{cases}
\tag{2}
$$

where

$$
\begin{cases}
\Delta f^i(k) = f(x^i(k),k) - f(x^{i-1}(k),k) \\
\Delta B^i(k) = B(x^i(k),k) - B(x^{i-1}(k),k) \\
\Delta x^i(k) = x^i(k) - x^{i-1}(k)
\end{cases}
\tag{3}
$$

and $\omega^i(k) \in R^n$ is the undetermined factor vector; $\upsilon^i(k) \in R^r$ is the noise disturbance vector; $x^0(0) \ne x_d(0)$ is the initial state vetor.

The experimental results show that the muscle contraction force grows with the stimulation strength. Therefore, the magnitude of the stimulation pulse trains should be limited to some extent with a view to application. For the i th iteration, the output vector on the basis of the saturation function $sat(\cdot)$ in (1) are defined as following,

$$
u^i(k) = sat(\overline{u}^i(k))
\tag{4}
$$

where

$$
sat(\overline{u}^i(k)) = [sat(\overline{u}_1^i(k)), sat(\overline{u}_2^i(k)), \cdots, sat(\overline{u}_m^i(k))]^T
$$
, and

$$
sat(\overline{u}_j^i(k)) =
\begin{cases}
\overline{u}_j^i(k), & if \left| \overline{u}_j^i(k) \right| \le (\overline{u}^i)^* \\
\dfrac{\overline{u}_j^i(k)}{\left| \overline{u}_j^i(k) \right|} (\overline{u}^i)^*, & if \left| \overline{u}_j^i(k) \right| > (\overline{u}^i)^*
\end{cases}
\tag{5}
$$

for $j = 1,2,\cdots,m$, $(\overline{u}^i)^* > 0$.

The sum of the output vector is

$$
\overline{u}^i(k) = u_{il}^i(k) + u_{fw}^i(k)
\tag{6}
$$

where $u_{il}^i(k)$ and $u_{fw}^i(k)$ are the iterative learning controller output vector and the forward controller output

vector, respectively.

The iterative learning controller is described by

$$
z_{il}^i(k+1) = h_a(z_{il}^i(k)) + H_b(z_{il}^i(k))e^i(k)
\tag{7}
$$

$$
u_{il}^i(k) = h_c(z_{il}^i(k)) + H_d(z_{il}^i(k))e^i(k)
\tag{8}
$$

where $e^i(k) = y_d(k) - y^i(k) \in R^r$ is the output error vector; $y_d(k) \in R^r$ is the desired output vector; $y^i(k) \in R^r$ is the real output vector; $z_{il}^i(k) \in R^{n_c}$ is the state vector of the feedback controller, and $z_{il}^0(0) = 0$; function vector $h_a(\cdot): R^{n_c} \mapsto R^{n_c}$ and $h_c(\cdot): R^{n_c} \mapsto R^m$ are bounded, i.e.

$$
\begin{cases}
\left\| h_a(z_{il}^i(k)) \right\| \le b_{h_a} \left\| z_{il}^i(k) \right\| \\
\left\| h_c(z_{il}^i(k)) \right\| \le b_{h_c} \left\| z_{il}^i(k) \right\|
\end{cases}
\tag{9}
$$

The function matrix

$H_b(\cdot): R^{n_c} \mapsto R^{n_c \times r}$ and $H_d(\cdot): R^{n_c} \mapsto R^{m \times r}$ are bounded, i.e. $\forall k \in [0,N], \forall z_{il}^i(k) \in R^{n_c}$

$$
\begin{cases}
\left\| H_b(z_{il}^i(k)) \right\| \le b_{H_b} \\
\left\| H_d(z_{il}^i(k)) \right\| \le b_{H_d}
\end{cases}
\tag{10}
$$

where $b_{h_a}, b_{h_c}, b_{H_b}, b_{H_d}$ are the constant and greater than zero.

Adaptive iterative learning algorithm associated with the feedback controller is given by

$$
u_{fw}^{i+1}(k) = \overline{u}^i(k) + \sum_{l=1}^{M} \widetilde{K}_l(k)e^{i-l+1}(k+1)
\tag{11}
$$

where $M \ge 1$ is the iteration numbers, and $\widetilde{K}_l(k) \in R^{m \times r}$ is the learning matrix.

The adaptive iterative learning control architecture is presented as Fig. 3. The system consists of two parts of the iterative learning controller and the forward controller. The total control signals are generated only by the forward controller before the iterative learning controller works. At the same time, both the feedback error signals and the previous output stimulation signals are put into the iterative learning controller, and are

processed according to some control rules. As the time increases, the feedback errors decrease and then the iterative learning controller plays the leading adjustment role in the system. The one advantage is that learning and controlling carry out simultaneously, and the other is that the previous feedback errors and previous stimulation signals are considered in the proposed control algorithm. It is very applicable to the nonlinear time-varying skeletal muscle systems.
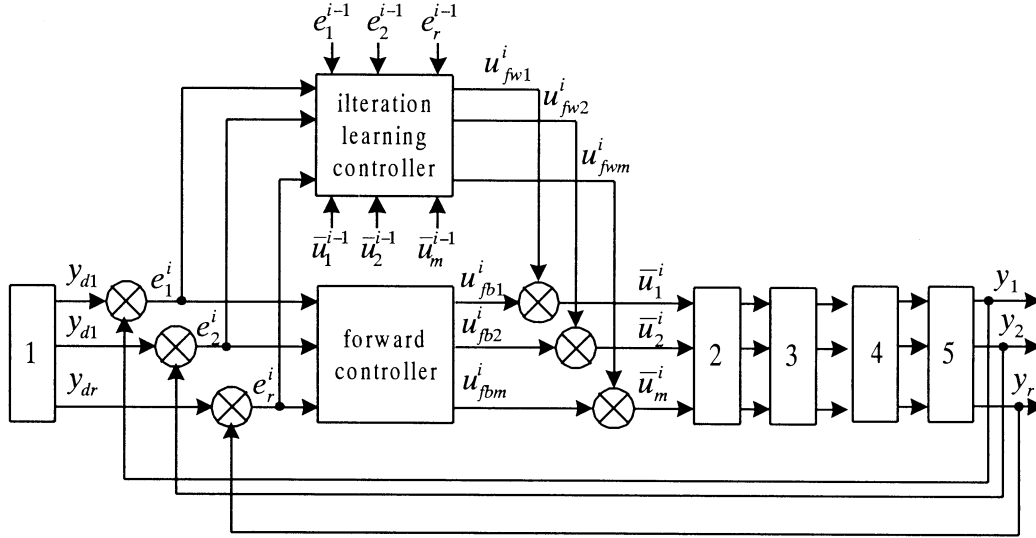
1.motion patterns; 2.stimulation patterns; 3.stimulators; 4.extensor and flexor; 5. musculoskeletal

Fig. 3   Structure of the adaptive iteration learning controller for FNS system

4. Experimental results

For comparative purpose, two clinical experiments for applied for the elbow and the wrist flexion motion control with FNS are illustrated by two different control algorithms. First, we conduct the experiment with the conventional PID control algorithm and then with the iterative learning control algorithm above.

Fig. 4 shows the different results of both iterative learning controller and PID controller on the FNS multiarticular muscle control system. We can see that the real joint angle trajectories of the elbow flexion motion with PID control algorithm are characterized by sharp oscillation. Furthermore, the stimulating signal fluctuates severely. The clinical experimental results show that the patient does not feel very well or has obvious pains in the muscle when such stimulating signal obtained from PID algorithm is stimulated to the muscle. However, we see that the joint angle trajectories of the elbow flexion by mean of the iterative learning control algorithms are very close to the desired output trajectories. Also, the stimulating signal varies smoothly. In particular, the clinical experiments show that the patient has not any pains while the stimulating signal figured out with the proposed iterative learning control algorithm is acted on the skeletal muscles.
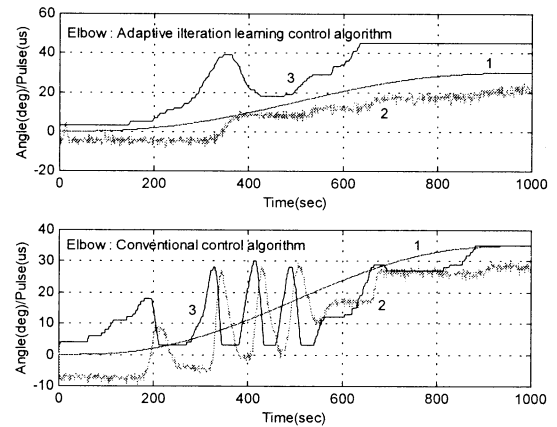
1. desirable elbow joint motion angular   2. Real elbow joint motion angular 3. Stimulating signals

Fig. 4   The trajectory-following curves of elbow flexion with different control algorithms
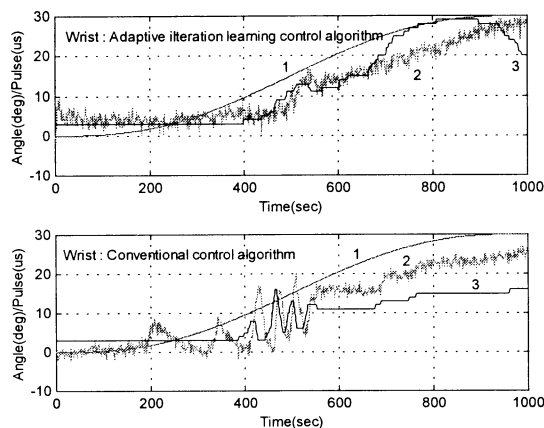
1. desirable wrist joint motion angular   2. Real wrist joint motion angular 3. Stimulating signals

Fig. 5   The trajectory-following curves of wrist flexion with different control algorithms

Fig. 5 shows the different results of both iterative learning controller and PID controller on the FNS multiarticular muscle control system. We can see that the real joint angle trajectories of the wrist flexion motion with PID control algorithm are characterized by sharp oscillation. Furthermore, the stimulating signal fluctuates severely. The clinical experimental results show that the patient does not feel very well or has obvious pains in the muscle when such stimulating signal obtained from PID algorithm is stimulated to the muscle. However, we see that the joint angle trajectories of the wrist flexion by mean of the iterative learning control algorithms are very close to the desired output trajectories. Also, the stimulating signal varies smoothly. In particular, the clinical experiments show that the patient has not any pains while the stimulating signal figured out with the proposed iterative learning control algorithm is acted on the skeletal muscles.

5.   Conclusions

(1)   The iterative learning control technique applied to FNS multiarticular muscle control system offers advantages over conventional PID control method. The joint angle trajectory-following of FNS multiarticular muscle system based on the iterative learning controller is satisfactory. Furthermore, the stimulated patient does not feel bad or has not any pains in the muscle when such stimulating signal generated by both iterative learning controller and feedback controller is activated to skeletal muscles.

(2)   Further research should be directed towards the relaxation restrictions or the elimination of the assumption on the initial conditions and generality of the iterative learning algorithm for limb motion control with functional neuromuscular stimulation.

References

[1]   Uchiyama M (1978), Formation of high-speed motion pattern of a mechanical arm by trial. Transactions of the Society of Instrumentation and Control Engineers, vol.14, pp.706-712

[2]   Arimoto S, Awamura K and Miyazaki F (1984), Bettering operation of robots by learning. Journal of Robotic Systems, vol.1, no.2, pp.123-140

[3]   Kawamura S, Miyazaki F and Arimoto S (1988), Realization of robot motion based on a learning method. IEEE Transactions on Systems, Man, and Cybernetics, vol.18, no.1, pp.126-134

[4]   Oh SR, Bien Z and Suh IH (1988), An iterative learning control method with application for the robot manipulator. IEEE Journal of robotics and automation, vol.4, no.5, pp.508-514

[5]   Bien Z and Huh KM (1989), Higher-order iterative learning control algorithm. IEE Proceedings-D, vol.136, no.3, pp.105-112

[6]   Heinzinger G, Fenwick D, Paden D and Miyazaki F (1992), Stability of learning control with disturbances and uncertain initial conditions. IEEE Transactions on Automatic Control, vol.37, no.1, pp.110-114

[7]   Wu H, Zhou Z and Xiong S (1999), Neural network feedback linearization control for the musculoskeletal movement with functional neuromuscular stimulation. In: Chen Z, Zhao G (ed), Proceedings of the 4th ASIAN/PACIFIC International Symposium on Instrumentation, Measurement & Automatic Control (IMC'99), Beijing: International Academic Publishers, 1999, pp.333-336

[8]   Wu H, Zhou Z and Xiong S (2000), Experimental research on motion trajectory-following of human multi-joint with FNS. Journal of Chinese Biomedical Engineering (accepted)

Visualisation of Evolutionary Algorithms using Principal Components Analysis

J.J. Collins and Michael O'Neill
Dept.of Computer Science and Information Systems
University of Limerick,
Limerick. Ireland.
Email: [j.j.collins, michael.oneill]@ul.ie

Abstract

Modelling the behaviour of genetic algorithms has concentrated on Markov chain analysis which yield little insight into the dynamics of the underlying processes. A framework for global modelling and visualisation of genetic algorithms is described based on Principal Component Analysis. An instance of a population is transformed into a compact low dimensional eigenspace representation, and a pattern vector calculated for each population of strings by projecting it into the eigenspace. A 3D manifold or global is derived from the set of computed pattern vectors, and possible interpretations of the derived results discussed.

1 Introduction

High degrees of consonance correspond to the concept of a GA-easy problem [3], where consonance is a measure of the correlation between the problem domain and the GA parameters ie. genetic operator, probability factors and genocode. Our objective is to derive a methodology and associated tools for modelling and measuring the dynamic behaviour of the underlying GA process. Traditionally, Markov chain analysis techniques [8] have formed the significant tools for analysis into the long term, steady state behaviour of large populations of GAs. However, within practical constraints, Markov chain analysis yields little information on the transient behaviour of the GA. The authors subscribe to the view that in modelling consonance/disconsonance, one needs to elicit and measure the dynamics as opposed to the long term behaviour.

Principal Components Analysis (PCA) is used as the methodology for deriving a model of the transient behaviour of a GA. This approach is based on extracting the significant global features of the data set, and encoding them in a more compact form. In computing the principal components, the eigenvectors for a set of generations are derived [5, 9, 10, 1], which constitute the dimensions of the eigenspace for that particular data set. Each generation is then projected into the eigenspace to derive a weight vector. Plotting the first 3 elements of each weight vector yields a manifold, parameterised by the encoding scheme used and the problem domain. This manifold describes the global dynamic signature of the GA process based on the evolved populations over time [7].

2 PCA

Eigenspaces use global statistical data to encode the *relevant* content of a data instance, by deriving the principal components of the distribution of a set of such instances. The principal components or eigenvectors of the covariance matrix are derived using the using the Karhunen-Loève transform. [5, 9]. The resultant eigenvectors can be thought of as a set of features that together characterise the global statistical variations amongst the data set. The eigenvectors constitute the dimensions of an eigenspace for the specific data set. These eigenvectors form an orthogonal basis set for generating a compact representation of the original data set.

Principal Components Analysis (PCA), has been applied in the field of computer vision for statistical modelling and classification. In the context of face recognition, the data set consists of a set of face images and the eigenvectors are known as eigenfaces [10]. Each face can be represented exactly in terms of a linear combination of these said eigenfaces. For a set of M face images that have a unified size N, where N = width x height of image, each image is represented as a vector in an $N-$dimensional vector space (hyperspace). Using PCA, M' eigenvectors are calculated yielding an M' dimensional eigenspace, where
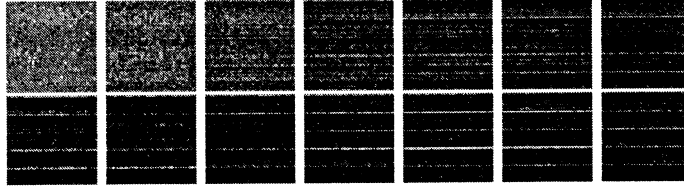
Figure 1: For F3 using E-code with 50 alleles per generation: Top row shows generations 1, 3, 5, 10, 15, 20 and 25. Generations 50, 75, 100, 125, 150, 175 and 200, are depicted on the bottom row.

Figure 2: For F8 using E-code: the average generation image (right), and the first five most significant eigenimages from left to right.

$M' \leq M - 1$. The M vectors are then projected into this low dimensional subspace. In the case of recognition, classification is performed by finding a match that minimises the Euclidean distance between the input face image projected into the subspace and a face class represented in this eigenspace.

Given a set of images (snapshot of a population at time t): $\Gamma_1, \ldots, \Gamma_M$; an average image can be computed as $\Psi = \frac{1}{M} \sum \int_{i=1}^{M} \Gamma_i$ A new set $\Phi_1, \ldots, \Phi_M$ is given by $\Phi_i = \Gamma_i - \Psi$ which translates the original images by Ψ in the image space. The principal components of the new space given by Φ_i are the eigenvectors of its covariance matrix:

$$C = \frac{1}{M} \sum_{i=1}^{M} \Phi_i \Phi_i^T = AA^T \qquad (1)$$

where $A = [\Phi_1 \ldots \Phi_M]$.

Singular value decomposition is used to reduce the matrix size:

$$(AA^T)(AV_i) = \lambda_i(AV_i) \qquad (2)$$

where (AV_i) are the eigenvectors of AA^T, and λ_i are the corresponding eigenvalues. The eigenvectors of the covariance matrix C are AV_i. Therefore, the *full set* of eigenvectors of C are given by:

$$U_i = AV_i = \sum_{k=1}^{M} v_k^i \Phi_k \qquad (3)$$

where $i = 1, \ldots, M - 1$ and v_k^i is the kth element of V_i. For a given set of eigenvectors $\{U_k\}$, an image Γ is projected onto the eigenvectors by:

$$\omega_k = \frac{U_k^T(\Gamma - \Psi)}{\lambda_k} \qquad (4)$$

where $k = 1, \ldots, M'$, $M' \leq M - 1$. A weight distribution vector or pattern vector is given by $\Omega = [\omega_1 \ldots \omega_{M'}]$. For a new image image, one can calculate the Euclidean distance ε between its pattern vector and the average pattern vector of a known image class (Ω_c) by $\varepsilon = \| \Omega - \Omega_c \|$.

3 Results

Test function F1 to F5 are taken from De Jong [2]. Functions F6 (Rastrigin), F7 (Schwefel), F8 (Griewangk) are taken from *Mühlenbein et al.* [6], F9 and F10 from *Schaffer et al.*, F11 from Ackley, and F12 is a modified version of Himmelbalu's function taken from *Deb*.

3.1 Populations as Images

For each of the 12 problem domains $\{F1, \ldots, F12\}$, further differentiated by one of the three coding schemes – binary, E-code and Gray – twenty five trial runs were generated. Each run was terminated when 200 generations had been evolved. Each generation or population instance was converted into raster image format and scaled [1] to 256 x 256. Thus, in total, thirty six data sets were generated where each data set contained 5,000 images. For each data set, a mean set of population images was calculated.

Fig.1 depicts mean population images computed for test function F8 using E-code. The x-axis of an image represents the population size which was set at fifty, and the y axis is a chromosome locus for an allele. A

---

[1] Scaling was necessary to facilitate correlation of derived eigenspace representation from different domains.
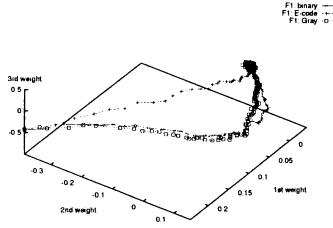
Figure 4: For F1-binary, and F1-Ecode and F1-Gray projected into its eigenspace.

Figure 5: Ccorrelation landscape.

black pixel represents a chromosome locus value of 0, white for 1, and the various shades of grey represents values that are ambiguous ie $\in \{0, 1\}$. Capturing the essence of a population as an image facilitates visual observation of propagation of useful schemata from preceding to proceeding generations.

3.2 Eigenimages

In computing the eigenimages of a distribution of populations, one derives the major axes of the distribution (the principal components), based on the conceptual idea of a population as a point in a hyperspace. For the set of mean populations images derived, applying PCA with $M' = 10$ will yield the most significant ten eigenvectors. In practice, only the first three eigenvectors are needed to generate a manifold. Fig. 2 deppicts the results of applying PCA to the mean population set of problem domain F8 using E-code to encode the genotype.

3.3 Manifolds

Pattern or weight vectors are derived for each image by projecting it into its corresponding eigenspace, to yield the expansion coefficients. A 3D manifold is derived by plotting the first three elements of every pattern vector calculated for every image in the data set [7]. Manifolds can be conceptualised as a global signature of the dynamics of the underlying process. An example manifold for F2 is depicted in fig.3. Even though all computed manifolds are parameterised by different test functions and encoding schemes, they all display a relatively uniform curve in the x-y axis but vary in the z-axis. Each point p on the manifold, represents the relationship of a population p with all other populations generated. For each manifold shown, generation 1 is represented as the first point on the manifold curve starting at the upper corner of the plot. In general, the E-code manifolds are smoother and have
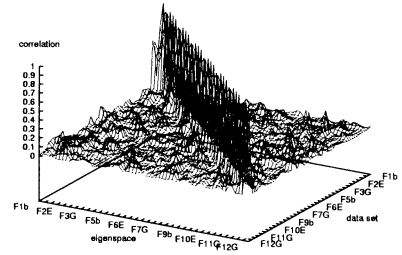
a more regular distribution than either their binary or Gray counterparts.

3.4 Correlation of Manifolds

Each eigenspace is analogous to a specific coordinate system based on the data set on which the Karhunen-Loève transform was applied. To evaluate the effects of encoding schemes on the resultant manifolds, for test function F1, all three data sets were projected into the three eigenspaces, as specified by binary, E-code and Gray representations. Fig.4 illustrates the resultant similarity between binary and Gray manifolds, but the E-code manifold does not have a close correlation. For other test functions the same procedure resulted in projected manifolds that were very disimilar. To evaluate the uniqueness of each manifold, as parameterised by a specific function domain and encoding scheme, each of the thirty six data sets was projected into a specific eigenspace, and manifolds calculated. Each manifold was correlated with the manifold of the data set used to compute the said eigenspaces. This process was repeated for all thirty six eigenspaces to yield a correlation landscape. The correlation measure used was simple linear regression, and lies in the range of $-1 \leq C \leq 1$. A reasonable rule of thumb of thumb is to assume that the correlation is weak if $0 \leq |C| \leq 0.5$, strong ifif $0.8 \leq |C| \leq 1$, and moderate otherwise. The correlation landscapes as illustrated in fig.5, emphasise the uniqueness of manifolds as parameterised by a problem domain. This is evidenced by the diagonal traversing a flat surface. The diagonal represents the correlation of a manifold with itself which is 1, or a weak to moderate correlation between manifolds from the same problem domain but using different genotype encodings.
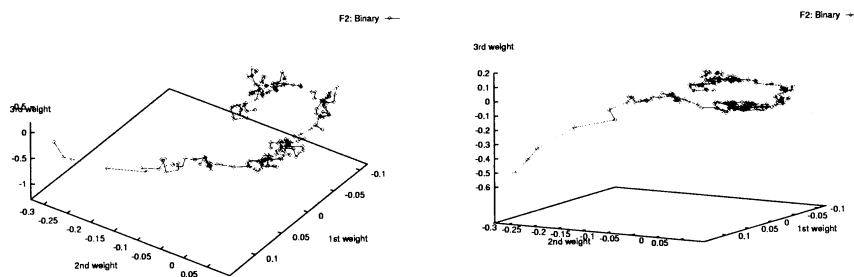
Figure 3: GA manifold parametrised by test function and coding schemes - for F2 using binary (right), the middle depicts the same manifolds as on the right , but using a different viewing angle.

4 Conclusion

As a first step in deriving a methodology and associated tool for modelling the GA process, PCA has yielded promising results. The task of converting a set of data instances of generations, into raster format, may yield the evolving schemata upon visual inspection. This is a task which is amenable to traditional image processing techniques such as edge detectors and segmentation using region reconstruction from borders. The manifolds derived are regular with variations along the xy and xz axis, which indicates that the Karhunen-Loève transform (PCA), with further work, may be of value as a tool in modelling the dynamic characteristics of GAs.

The manifold is a compact representation or a dynamic fingerprint for the GA process. The modelling tool described has potential in signalling when a GA is trapped at local optimum or reached a global optimum, by identifying when the evolving manifold curve becomes trapped in a local area or is static. Analysis of the efficiency of the GA process could be evaluated by measuring the regularity of the distribution of points on the curve. The E-code manifolds derived are generally smoother and have a more regular distribution than either their binary or Gray counterparts. Manifolds can now be used as a tool for evaluating the effects of various operators (ie. crossover - 1-point, 2-point, uniform) on the overall performance of the GA.

References

[1] Collins, J.J., and Eaton, M. (1997). A Global Representation Scheme for Genetic Algorithms. In *Proc. Int. Conf. on Computational Intelligence, Dortmund, Germany*, pages 1–15. Springer LNCS 1226.

[2] De Jong, K. A. (1975). An Analysis of the Behaviour of a Class of Genetic Adaptive Systems. Phd thesis, Dept. of Computer and Communication Sciences, University of Michigan, Urbana-Champaign, IL.

[3] Spears, W.M., De Jong, K. A., and Gordon, D. F. (1994). Using markov chains to analyze gafos. In *Foundations of Genetic Algorithms:3, Estes Park, CO*, pages 115–137.

[4] Hinterding, R., Gielewski, H., and Peachy, T. C. (1995). The nature of mutation in genetic algorithms. In *Proc. of the Sixth Int. Conf. on Genetic Algorithms, University of Pittsburg*, pages 65–72.

[5] Kirby, M., and Sirovich, L. (1990). Applications of the Karhunen–Loéve procedure for the characterization of human faces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(1):103–108.

[6] Mühlenbein, H., Schomisch, M., and Born, J. (1991). The parallel genetic algorithm as function optimizer. In *Proc. of the Fourth ICGA*, pages 271–278.

[7] Murase, H., and Nayar, S. K. (1995). Visual learning and recognition of 3–D objects from appearance. International Journal of Computer Vision, 14:5–24, 1995.

[8] Nix, A. E., and Vose, M. D. (1992). Modeling GAs with markov chains. Annals of Mathematics and Artificial Intelligence, 5:79–88.

[9] Oja, E. (1983). Subspace Methods for Pattern Recognition. Research Studies Press, Herts.

[10] Turk, M., and Pentland, A. (1991). Eigenfaces for recognition. Journal of Cognitive Neuroscience, 3(1):71–86, 1991.

Comparison Between Synchronous and Asynchronous Implementation of Parallel Genetic Programming

Shisanu Tongchim
Department of Computer Engineering
Chulalongkorn University
Bangkok 10330, Thailand
g41stc@cp.eng.chula.ac.th

Prabhas Chongstitvatana
Department of Computer Engineering
Chulalongkorn University
Bangkok 10330, Thailand
prabhas@chula.ac.th

Abstract

An evolutionary method such as Genetic Programming (GP) can be used to solve a large number of complex problems in various application domains. However, one obvious shortcoming of GP is that it usually uses a substantial amount of processing time to arrive at a solution. In this paper, we presented the parallel implementations that reduced the processing time by using a coarse-grained model for parallelization and an asynchronous migration. The problem chosen to examine the parallel GP was a mobile robot navigation problem. The experimental results showed that superlinear speedup of GP was achieved.

1 Introduction

Genetic Programming was successfully used to perform automatic generation of mobile robot programs [1]. The use of the perturbation to improve robustness of the robot programs was proposed. Each robot program was evaluated under many environments that were different from the original one. As a result, the substantial processing time was required to evaluate the fitness of the robot programs.

To reduce the computational time, this study proposed two parallel implementations. Asynchronous and synchronous parallelization approaches were examined. We also compared the quality of the solutions generated from the serial and parallel GP.

The earlier work of parallel GP was implemented on a network of transputers by Koza and Andre [2]. Their result showed that the parallel speedup was greater than linear. Dracopoulos and Kent [3] proposed the use of the Bulk Synchronous Parallel Programming (BSP) model to parallelize genetic programming. Two approaches of parallel GP were examined on a cluster of Sun workstations. The first was based on a master-slave model while the second was based on a coarse-grained model. The results showed that the achieved speedup was close to linear. A recent paper by Punch [4] presented the empirical study about some problem-specific factors which affect the effectiveness of parallel GP. Punch concluded that the achieved performance of parallel GP by using a coarse-grained model may vary according to the nature of problems.

The remaining sections are organized as follows: The next section is a description of the mobile robot navigation problem. Section 3 describes a problem representation in the serial GP. Section 4 shows the parallel solutions. Section 5 presents the experimental results and discussion. Finally, section 6 provides the conclusions of this work.

2 Mobile Robot Navigation Problem

Our previous work [1], GP was used to generate a robot control program for the mobile robot navigation problem. The task was to control a mobile robot from a starting point to a target point in a simulated environment. The environment was filled with the obstacles which had several geometrical shapes.

The aim of the work was to generate *robust* control programs. In the evolution process, each individual was evaluated under many environments that were different from the original one. The result showed that the robustness of the robot programs was improved by such an approach.

3 Serial Algorithm

The terminal set is composed of three primitive movement controls {move, left, right} and one sensor information {isnearer}. The function set is composed of three functions {if-and, if-or, if-not}. The GP parameters are shown in Table 1.

The fitness function is a sum of the fitness value in each environment which is based on the distance of the final position and the number of moves.

In the evolution process, the percent of disturbance is 20% and the number of training environments is 8.

Table 1: GP parameters

Total population	6000
Crossover probability	0.9
Mutation probability	0.1
Reproduction	5% of Total population
Maximum generation	200

Table 2: Experimental Parameters

	Num. of Processors				
	1	2	4	6	10
Pop. size *	6000	3000	1500	1000	600
Environments *	8	7	4	3	2
Migration interval	NA	100	50	34	20

* per node

4 Parallel Genetic Programming

In a coarse-grained model, the population is divided into subpopulations and these subpopulations are maintained by different processors. The model is also known as *Island model* and the subpopulation is called *deme* [2].

Some works in parallelization of GA and GP using a coarse-grained model [2, 5] showed that the results achieved *superlinear speedup*[1]. This is caused by two factors; the speedup from the populations distributed across different processors and the speedup obtained by increasing the probability in finding the correct solution, as the number of populations is increased.

In applying the parallel approach to the previous work [1] by using a conventional coarse-grained model, the result achieves only linear speedup [6] since the amount of work is fixed – the algorithm is terminated when it reaches the maximum generation. Hence, the parallel algorithm does not exploit the probabilistic advantage that the answer may be obtained before the maximum generation. We reduce redundant jobs by dividing the *environments* among the processing nodes. After a specific number of generations, all subpopulations are migrated between processors using a fully connected topology. However, this scheme leads to the reduction of robustness since each individual has a shorter period in each training environment. To mend this problem, we increase the number of environments in each node. However, the number of environments per node should be less than the number of environments per node in the general coarse-grained model.

We implement our parallel algorithm on a dedicated cluster of PC workstations with 350 MHz Pentium II processors, each with 32 Mb of RAM, and running Linux as an operating system. These machines are connected via 10 Mbs ethernet cabling. We extend the program used in [1] to run under a clustered computer by using MPI as a message passing library.

Several trials are examined to find an appropriate value for the number of environments per node (see Table 2). The migration is carried out as follows: each

[1]*Superlinear speedup* means that speedup is greater than the number of processors used.

```
procedure Migration
begin
    barrier¹ wait all nodes ready
    for i = 1 to n
    begin
        if (my_process_id = i)
            broadcast send
        else
        begin
            broadcast receive
        end
        barrier² wait for the next broadcast
    end
end
```

Figure 1: The migration process

node broadcasts its subpopulation to all other nodes by MPI_Bcast function, this is repeated for every node. The top 5% of individuals from each subpopulation are exchanged during the migration. Pseudo-code for the migration is shown in figure 1. The detail will be discussed in the timing analysis section.

The total population is held constantly for the task and is divided equally among workstations. The number of selected individuals, crossover operation, mutation operation, reproduction are a percentage of the amount of the total population. The parallel efficiency is measured by varying the number of nodes and the results are averaged over 20 runs for each number of nodes.

In the first implementation, the migration between subpopulations is synchronized. Each node is blocked by MPI_Barrier function until all subpopulations evolve to the same number of generations. However, the synchronizing migration results in uneven work loads among the processors since the time required to complete the evaluation varies, with the least effective programs taking the longest period and the best programs taking the shortest period.
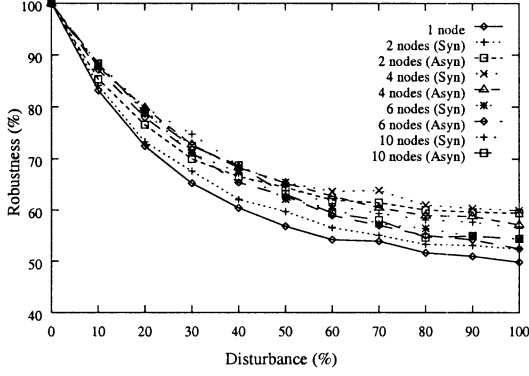
Figure 2: Robustness

Figure 3: Speedup

In the second implementation, we attempt to further improve the speedup of the parallel algorithm by the asynchronous migration. When the fastest node reaches predetermined generation numbers, the migration request is sent to all subpopulations. The migration takes place at the end of the current generation. In this state, if any populations are still in the fitness evaluation phase, the other nodes must wait. The waiting time will be at most less than one generation.

5 Results and Discussion

5.1 Speedup

To make an adequate comparison between the serial algorithm and parallel algorithm, Cantú-Paz [7] suggests that the two must give the same quality of the solution. In this paper, we define the robustness of the generated program from the serial algorithm as a baseline. In addition, if the generated program from the parallel algorithm gives the same robustness as the program from the serial algorithm, the equal amount of work to achieve the same quality of the answer is done. From the robustness graph in figure 2, the generated program from the parallel GP is better than the serial GP. Hence, the amount of work from the parallel algorithm is not less than the serial algorithm.

The parallel speedup is defined as the ratio of the serial execution time to the parallel execution time.

$$Speedup = \frac{Serial\ time}{Parallel\ time} \tag{1}$$

Figure 3 illustrates the speedup observed on the two implementations as a function of the number of processors used. The performance is less than we expect, although both implementations exhibit superlinear speedup. The speedup curves taper off for 10 processors and the performance of the asynchronous implementation is slightly better than the performance of the synchronous implementation. In order to discern the cause
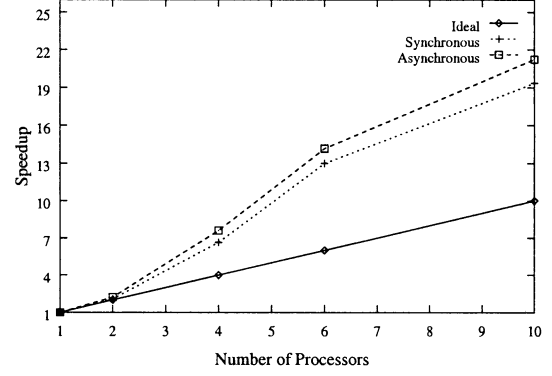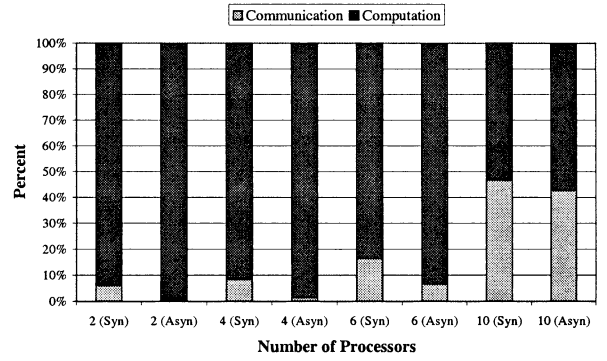
Figure 4: Percentage of time spent in computation and communication

of this result, the timing analysis is performed in the next section.

5.2 Timing Analysis

Figure 4 shows the relative time spent in each section of the implementations. The communication overhead – the sum of the barrier time and broadcast time – goes up considerably as the number of processors increases. The asynchronous implementation does not help much on reducing the communication overhead at large numbers of processors. Thus, we investigate the further detailed analysis of the communication overhead.

Figure 5 shows the absolute time spent in major functions of the communication. The time spent in barriers indicates the time spent on waiting for all processes to reach the same point. From pseudo-code of the migration in figure 1, the barrier time consists of the time spent to wait for all nodes to be ready for the migration and the next broadcast.

In the synchronous implementation, the time spent in barriers reduces as the number of processors increases. This is because the barrier time depends on the variation
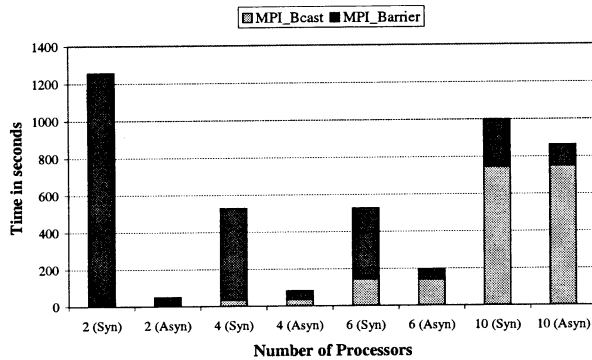
Figure 5: Absolute time spent in communication

of the computation time of each node. As the number of nodes is increased, the computation time per node is decreased. Hence, the barrier time is reduced.

In contrast, the barrier time in the asynchronous implementation increases as the number of processors increases. This is due to the fact that the time spent in the second barrier (waiting for the next broadcast) increases with the number of nodes. However, the asynchronous implementation eliminates the first barrier therefore it reduces the total time in the barriers compared to the synchronous implementation.

The absolute time spent in a broadcast increases considerably – greater than linear. By inspecting in the trace information using a visualization tool, we found that the transmission of the broadcast functions in the implementation of MPI that we use may be executed more than once, especially for a large number of processors.

After obtaining some timing analyses, the results reveal the cause of the problem. The performance degradation in 10 processors is caused by the excessive communication time due to the broadcast function. Although the asynchronous migration reduces the barrier time effectively compared to the synchronous migration, the increase in the communication time in 10 processors obliterates this advantage. In case of a small number of processors (2,4,6), the gain from the asynchronous migration is considerable as the evolution proceeds at the speed of the fastest node.

As the size of the work increases (i.e., the number of training environments increases), the serial and parallel computation time will be increased when the time spent in the communication is constant. If the ratio of the computation/communication can be kept large (large work load), then one can expect that the parallel performance will be improved.

6 Conclusions

The result presented in this paper shows a success of speeding up the Genetic Programming process by means of parallel processing. The parallel implementations of Genetic Programming successfully exploit the computing resource of a dedicated cluster of PC workstations. Superlinear speedup of GP can be acquired by improving a coarse-grained model for parallelization as less computational work needs to be done. Furthermore, the timing analyses indicate the scalability of the parallel approaches, as the size of the problem increases, the speedup will be improved.

References

[1] Chongstitvatana P (1998), Improving robustness of robot programs generated by genetic programming for dynamic environments. Proc. of IEEE Asia Pacific Conference on Circuits and Systems, p.523–526

[2] Koza JR, Andre D (1995), Parallel genetic programming on a network of transputers. Proc. of the Workshop on Genetic Programming: From Theory to Real-World Applications, University of Rochester, National Resource Laboratory for the Study of Brain and Behavior, Technical Report 95-2, p.111-120

[3] Dracopoulos DC, Kent S (1996), Bulk synchronous parallelisation of genetic programming. Proc. of the Third International Workshop on Applied Parallel Computing in Industrial Problems and Optimization (PARA '96), Springer Verlag, Berlin

[4] Punch B (1998), How effective are multiple populations in genetic programming. Proc. of the Third Annual Conference in Genetic Programming, pp.308-313

[5] Lin S-C, Punch WF, Goodman ED (1994), Coarse-grain parallel genetic algorithms: Categorization and new approach. Proc. of the Sixth IEEE SPDP, pp.28-37

[6] Tongchim S, Chongstitvatana P (1999), Speedup Improvement on Automatic Robot Programming by Parallel Genetic Programming. Proc. of 1999 IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS'99), Phuket, Thailand

[7] Cantú-Paz E (1999), Designing efficient and accurate parallel genetic algorithms. PhD thesis, University of Illinois at Urbana-Champaign

A multi-agent reinforcement learning method
with the estimation of the other agent's actions

Yasuo Nagayuki[†1]   Shin Ishii[†1†4†5]   Minoru Ito[†1]   Katsunori Shimohara[†2†4]   Kenji Doya[†3†5†1]

†[1] Graduate School of Information Science, Nara Institute of Science and Technology
8916-5, Takayama-cho, Ikoma-shi, Nara 630-0101, Japan

†[2] NTT Communication Science Laboratories
2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan

†[3] ATR International
†[4] ATR Human Information Processing Research Laboratories
†[5] CREST, Japan Science and Technology Corporation
2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan

Abstract

The application of reinforcement learning to multi-agent systems has attracted recent attention. In a multi-agent environment, whether one agent's action is good or not depends on the other agents' actions. In traditional reinforcement learning methods, which are based on a stationary environment, it is hard to take account of the other agent's actions which may dynamically vary due to learning. In this article, we consider a two-agent cooperation problem, and propose a multi-agent reinforcement learning method based on the estimation of the other agent's actions. In our learning method, one agent estimates the other agent's action based on the internal model of the other agent. The internal model is acquired by the observation of the other agent's actions. Through experiments, we demonstrate that good cooperative behaviors are achieved by the use of the internal model of the other agent.

keywords: reinforcement learning, Q-learning, multi-agent systems, internal model, action estimation, cooperation problem.

1   Introduction

The realization of behaviors in multi-agent systems is an interesting topic from the viewpoint of engineering and cognitive science. In particular, reinforcement learning [1] of behaviors has attracted recent atten-

tion because of its adaptability to dynamic environments. Reinforcement learning has been applied to multi-agent problems such as pursuit games [2, 3, 4], soccer [5, 6, 7], prisoner's dilemma games [8], and coordination games [9, 10, 11].

Traditional reinforcement learning methods have been developed in a single-agent environment which is modeled as a Markov decision process (MDP). In an MDP, the transition of an environment's state is defined by a transition probability function that does not change with time. When considering a multi-agent environment in which the agents autonomously learn, the transition function often changes with time, because the policies of the other agents, which are regarded as a part of the environment, change according to their learning. Thus, the environment cannot be modeled as an MDP. In many multi-agent reinforcement learning studies, however, reinforcement learning methods based on the MDP are applied without much modification. In other words, those approaches ignore the difference between learning agents and a stationary environment.

In this article, we propose a multi-agent reinforcement learning method that explicitly takes the policy of the other agents into account. We consider a multi-agent environment with two agents and assume that they synchronously execute their actions and there is no inter-agent communication except for the observation of each other's actions. In our learning method, each agent learns to take actions based

on the estimation of the other agent's action. For this purpose, each agent updates an internal model of the other agent's policy using the observation of the other agent's actions. We experimentally evaluate our learning method by applying it to a variant of the pursuit problem [12], which is a typical multi-agent cooperation problem.

2 Markov decision process and Q-learning

Our multi-agent reinforcement learning method is based on Q-learning [13]. Q-learning was originally defined to deal with an MDP, and hence it should be used in an environment including only one learning agent.

An MDP is defined by a set of finite states of the environment, S, and a set of finite actions, A. At each time step, an agent observes a state $s (\in S)$, executes an action $a (\in A)$ and receives a reward r from the environment. The state transition of the environment is modeled by a transition probability function:

$$P_{ss'}^a \equiv \Pr(s'|s,a), \qquad (1)$$

where $\Pr(s'|s,a)$ is the probability that the environment changes to a new state $s' (\in S)$ when the agent executes action a in state s. The reward r is a variable, possibly probabilistic, which depends only upon the current state s and action a.

Q-learning [13] is an incremental reinforcement learning method. According to Q-learning, the agent selects an action based on an action value function (called the Q-function), $Q(s,a)$, which defines the expected sum of the discounted reward attained when executing action $a (\in A)$ in $s (\in S)$, and the subsequent actions are determined by the current policy. The Q-function is updated using the agent's experience. The learning flow is as follows:

1. For the current state s, the agent executes an action $a_i (\in A)$ with the probability:

$$\pi[s, a_i] = \frac{e^{Q(s,a_i)/T}}{\sum_{a_k \in A} e^{Q(s,a_k)/T}}. \qquad (2)$$

T is the parameter called the temperature, which determines the randomness of the stochastic policy (2). Although several methods for selecting an action have been proposed, we adopt this Boltzmann selection method.

2. The agent executes action a selected in step 1, and the environment changes to a new state s'

according to $P_{ss'}^a$. After a reward r is given from the environment, the Q-function is updated as follows:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r + \gamma \max_{a' \in A} Q(s',a')), \qquad (3)$$

where $\alpha(0 \leq \alpha < 1)$ is the learning rate and $\gamma(0 \leq \gamma \leq 1)$ is the discount factor.

3. If the new state s' satisfies a terminal condition, then a single episode ends. Otherwise let $s' \rightarrow s$ and go back to step 1.

When the temperature T is very small and the environment is an MDP (namely, $P_{ss'}^a$ is fixed), the above Q-learning converges and the optimal Q-function is obtained [13].

3 Multi-agent reinforcement learning

In this study, we consider a multi-agent environment that consists of learning agents and a stationary environment. Such an environment cannot be formulated as an MDP, because the transition probability $P_{ss'}^a$ changes over time due to the learning of the other agents. In such a case, the original Q-learning is not appropriate. In this article, we intend to extend Q-learning so as to deal with such a multi-agent environment. As a step in this direction, we propose a multi-agent reinforcement learning method based on the estimation of the other agents' action which may dynamically vary due to learning.

We consider a multi-agent environment with two agents and assume that they synchronously execute their actions and there is no inter-agent communication except for the observation of each other's actions.

In order to explicitly express the other agent actions, the Q-function is rewritten as $Q(s, a_{self}, a_{other})$. Here, $s (\in S)$ is an environment state that involves the two agents, $a_{self} (\in A_s)$ is an action of the agent under consideration and $a_{other} (\in A_o)$ is the other agent's action. A_s and A_o are the sets of actions which are possible for the agent under consideration and the other agent, respectively.

3.1 Estimation of internal model

The new definition of the Q-function clarifies that a_{other} is a hidden variable in our environment. In our reinforcement learning method, a_{other} is estimated based on the internal model $I(s, a_{other})$ of the other agent's policy, which is estimated from the past observation of the other agent's actions.

When the agent observes that the other agent executes an action $a^*_{other}(\in A_o)$ in s, function $I(s, a_{other})$ is updated for every executable action $a_{other}(\in A_o)$ in s:

$$I(s, a_{other}) \leftarrow (1-\theta)I(s, a_{other}) + \begin{cases} \theta & (a_{other} = a^*_{other}) \\ 0 & (otherwise) \end{cases},$$
(4)

where $\theta(0 \leq \theta < 1)$ is the parameter that controls the effect of the actual action a^*_{other}. If the other agent's policy is fixed and parameter θ decreases as an inverse of the number of observations, $I(s, a_{other})$ converges to the empirical probability that the agent executes a_{other} in s [1]. Since the other agent's policy changes over time due to learning, the learning of $I(s, a_{other})$ puts emphasis on recent observation, by using a constant for θ.

3.2 Reinforcement learning process

Here, we propose a multi-agent reinforcement learning method based on an estimation of the other agent's policy. The learning flow is as follows:

1. The agent under consideration observes the current state s, estimates the other agent's action a_{other} by $I(s, a_{other})$, and selects an action a_{self} based on its stochastic policy:

$$\pi[s, a_{self_i}] = \frac{e^{\bar{Q}(s, a_{self_i})/T}}{\sum_{a_{self_k} \in A_s} e^{\bar{Q}(s, a_{self_k})/T}}$$
(5)

$$\bar{Q}(s, a_{self}) \equiv \langle Q(s, a_{self}, a_{other}) \rangle_{I(s, a_{other})}$$
$$= \sum_{a_{other} \in A_o} I(s, a_{other}) Q(s, a_{self}, a_{other}),$$
(6)

where $\bar{Q}(s, a_{self})$ is the expected value of the Q-function with respect to $I(s, a_{other})$.

2. The agent executes the action a_{self} selected in step 1. Here, the other agent synchronously executes an action a^*_{other}. The environment changes to a new state s' and the agent receives a reward r from the environment. After that, function $I(s, a_{other})$ is updated according to equation (4), and $Q(s, a_{self}, a_{other})$ is updated as follows:

$$Q(s, a_{self}, a^*_{other}) \leftarrow (1 - \alpha)Q(s, a_{self}, a^*_{other})$$
$$+ \alpha(r + \gamma \max_{a'_{self} \in A_s} \bar{Q}(s', a'_{self})). \quad (7)$$

3. If the new state s' satisfies a terminal condition, then a single episode ends. Otherwise let $s' \rightarrow s$ and go back to step 1.
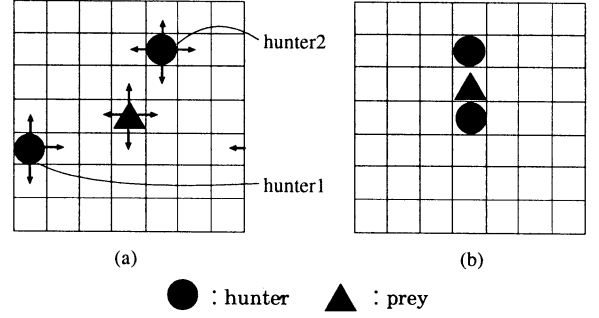
Figure 1: (a) Grid world of pursuit problem: Arrows show the directions which prey/hunters can move to, (b) An example of goal states.

4 Experiments and results

4.1 Pursuit problem

In order to evaluate our multi-agent reinforcement learning method, we prepared a variant of the well-known pursuit problem [12], which is a typical testbed for multi-agent cooperation algorithms.

- In a 7×7 toroidal grid world, two hunter agents and a prey exist, as shown in Figure 1(a). At the beginning of each episode, they are placed at random.

- At a discrete time step in an episode, the prey and the two hunters each synchronously executes one out of five actions: moving up, down, left, or right from the current position, or staying in the current position.

- The hunters perceive the current state, which is represented by the location of the prey relative to each hunter. For example, the state shown in Figure 1(a) is represented as $s = ([3, 1], [-1, -2])$, where, $[3, 1]$ and $[-1, -2]$ are the relative locations of hunter1 and haunter2 to the prey, respectively. The number of possible $s(\in S)$ is $2304(= 48 \times 48)$.

- When the two hunters capture the prey, the episode ends. The capture is achieved when the two hunters are positioned on both sides of the prey as shown in Figure 1(b).

- The prey's actions are independent of the hunters' positions. The probability of moving up, down or right is $\frac{1}{3}$, and the probability of the other actions, i.e., moving left or staying, is 0. This action definition implies that the state transition of this stationary environment is stochastic.
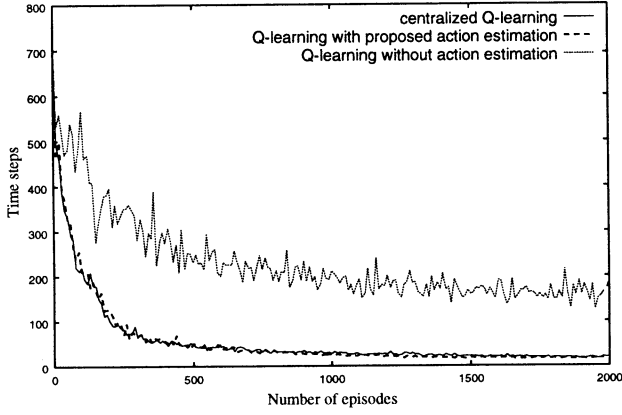
Figure 2: Time steps to capture the prey. Abscissa denotes number of learning episodes, and ordinate denotes average time steps to capture the prey. After every ten learning episodes, 100 evaluation episodes that vary the initial placements are done, and this figure shows their average.

4.2 Experimental results

Here, a "centralized Q-learning" (CQ) method is defined in order to evaluate our proposed learning method. In CQ, a centralized learning component is able to observe the whole grid world, and control the two hunter agents. The learning scheme is the usual Q-learning. Namely, its flow is almost identical to the one described in Section 2, except that an action is given by $a = (a^1, a^2)$, where, a^1 and a^2 are the actions of hunter1 and hunter2, respectively. In this CQ, the transition of the environment's states is stochastic, thus the environment can be represented as an MDP.

Figure 2 shows the average time steps needed to capture the prey. The parameters for the learning are: $\alpha = 0.3, \gamma = 0.9, T = 0.1$ and $\theta = 0.1$. The reward is designed such that if the two hunters capture the prey, then $r = 1.0$, else $r = 0$. The initial values of the Q-function are 0, and the initial values of $I(s, a_{other})$ are 0.2. In figure 2, the label "Q-learning without action estimation" (QwoAE) denotes a method in which $I(s, a_{other}) = 0.2$ for every s and a_{other}; namely, the estimation of the other agent's action is completely random. Figure 2 shows that the cooperative behaviors are well acquired by CQ or Q-learning with action estimation (QwAE), while QwoAE does not work well.

Figure 3 shows the average time steps needed to capture the prey where the two hunters have reward functions different from each other. One hunter re-
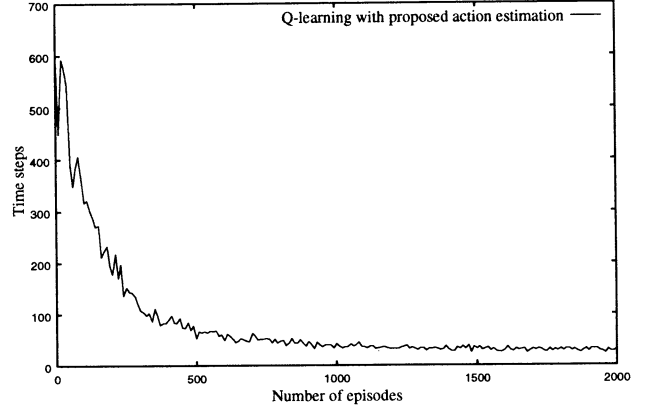
Figure 3: Time steps needed to capture the prey. One hunter receives $r = 1.0$ when capturing, $r = -0.01$ otherwise, and the other hunter receives $r = 0.5$ when capturing, $r = 0$ otherwise.

ceives $r = 1.0$ when it captures the prey, $r = -0.01$ otherwise. The other hunter receives $r = 0.5$ when capturing, $r = 0$ otherwise. The other parameters are the same as those in the previous experiment. It should be noted that it is difficult for CQ to deal with such heterogeneous agents.

5 Discussion

By comparing QwAE with QwoAE, the advantage of our action estimation method is obvious. In QwoAE, although the hunters learn to approach the prey, they do not learn cooperative capture movements. The capture thus occurs by chance.

By comparing CQ with QwAE, the learning performance of QwAE is similar to that of CQ. CQ is expected to learn effectively, because the environment can be represent as an MDP. In QwAE, on the other hand, there exists a hidden variable, i.e., the other agent's action, and the environment cannot be represented as an MDP. If the estimation of the hidden variable is not good, the learning will not proceed well, such as QwoQE. The similarity of the learning performance between CQ and QwAE implies that our estimation method works effectively.

In actual situations, the environment is often heterogeneous among agents. We believe such heterogeneousness induces self-organization in multi-agent systems. An important factor of this heterogeneity is the difference in rewards given to the agents. In central-

ized reinforcement learning, however, such an environment is hard to be considered. The result shown in Figure 3 implies that our new learning scheme is able to deal with such an environment.

6 Conclusion

In this article, we proposed a multi-agent reinforcement learning method based on an estimation of the other agent's action. In our learning method, one agent estimates the other agent's action based on the internal model of the other agent. The internal model is acquired by the observation of the other agent's actions. When applied to the pursuit problem, the experimental results showed that good cooperation behaviors were achieved by our new learning method. Furthermore, we showed that our new learning method could be applied to a heterogeneous environment, in which the rewards given to the agents were different from each other.

References

[1] R.S. Sutton, and A.G. Barto. *Reinforcement Learning: An Introduction*, MIT Press, 1998.

[2] S. Arai, K. Miyazaki, and S. Kobayashi. "Methodology in multi-agent reinforcement learning – approaches by Q-learning and profit sharing – ". *Journal of Japanese Society for Artificial Intelligence*, Vol.13, No.4, pp.609-618, 1998. (in Japanese)

[3] N. Ono, and K. Fukumoto. "Multi-agent reinforcement learning: a modular approach". In *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96)*, pp.252-258, 1996.

[4] M. Tan. "Multi-agent reinforcement learning: independent vs. cooperative agents". In *Proceedings of the Tenth International Conference on Machine Learning (ICML-93)*, pp.330-337, 1993.

[5] T. Balch. "Learning roles: behavioral diversity in robot teams". In *Collected papers from the AAAI-97 workshop on multiagent learning*, AAAI Press, 1997.

[6] M.L. Littman. "Markov games as framework for multi-agent reinforcement learning". In *Proceedings of the Eleventh International Conference on Machine Learning (ICML-94)*, pp.157-163, 1994.

[7] R. Salustowicz, M. Wiering, and J. Schmidhuber. "Learning team strategies: soccer case studies". *Machine Learning*, Vol.33, No.2/3, pp.263-282, 1998.

[8] T.W. Sandholm, and R.H. Crites. "Multiagent reinforcement learning in the iterated prisoner's dilemma". *Biosystems*, Vol.37, pp.147-166, 1995.

[9] C. Claus, and C. Boutilier. "The dynamics of reinforcement learning in cooperative multiagent systems". In *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98)*, pp.746-752, 1998.

[10] J. Hu, and M.P. Wellman. "Multiagent reinforcement learning: theoretical framework and an algorithm". In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML-98)*, pp.242-250, 1998.

[11] S. Sen, M. Sekaran, and J. Hale. "Learning to coordinate without sharing information". In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, pp.426-431, 1994.

[12] M. Benda, V. Jagannathan, and R. Dodhiawalla. "On optimal cooperation of knowledge sources ". *Technical Report* BCS-G2010-28, Boeing AI Center, 1985.

[13] C.J.C.H. Watkins, and P. Dayan. "Technical note Q-learning". *Machine Learning*, Vol.8, No.3, pp.279-292, 1992.

How Learning Can Affect the Course of Evolution in Dynamic Environments

Reiji SUZUKI   Takaya ARITA

Graduate School of Human Informatics, Nagoya University

Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan

E-mail: {reiji, ari}@info.human.nagoya-u.ac.jp

Abstract

The Baldwin effect is known as interactions between learning and evolution, which suggests that individual lifetime learning can influence the course of evolution without the Lamarckian mechanism. Our concern is to consider the Baldwin effect in dynamic environments, especially when there is no explicit optimal solution through generations and it depends only on interactions among agents. We adopt the iterated Prisoner's Dilemma as a dynamic environment and introduce phenotypic plasticity to strategies by using a meta-learning rule termed "Meta-Pavlov". In this simulation, the Baldwin effect was observed as follows: First, strategies with enough plasticity spread, which caused a shift from defective population to cooperative. Second, these strategies were replaced by the strategy [x00x], which has a modest amount of plasticity.

Keywords: baldwin effect, learning and evolution, the iterated prisoner's dilemma, artificial life.

1 Introduction

There have been a lot of discussions about interactions between learning and evolution. The Baldwin effect [1] is one of them, which suggests that individual lifetime learning can influence the course of evolution without the Lamarckian mechanism. This effect has come to the attention recently not only of biologists, but also of the computer scientists with the evolutionary simulation of Hinton and Nowlan [2]. Since Hinton and Nowlan, many studies have been conducted, most of which have discussed the effect on the assumption that environments are static and the optimal solution is fixed.

However, as we see in the real world, learning could be more effective and utilized in dynamic environments, because the flexibility of plasticity itself is advantageous to adapt ourselves to the changing world. Therefore, it is very important to examine how learning can affect the course of evolution in dynamic environments.

Our objective is to clarify the function and the mechanism of the Baldwin effect in dynamic environments focusing on balances between benefit and cost of learning, while most of the studies concerning the Baldwin effect have aimed at the static environments. In general, dynamic environments can be divided typically into the following two types: the environments in which the optimal solution is changed as the environment changes, and the ones in which each agent's fitness is decided by interactions with other agents.

As the former type of environments, Anderson [3] quantitatively analyzed how learning affects evolutionary process in the dynamic environment whose optimal solution changes through generations. Sasaki and Tokoro [4] studied the relationship between learning and evolution using a simple model, where individuals learn to distinguish poison and food by modifying the connective weights of neural network. These studies emphasized the importance of learning in dynamic environments.

We adopted the iterated Prisoner's Dilemma (IPD) as the latter type of environments, where there is no explicit optimal solution through generations and fitness of agents depends only on interactions among them. This paper describes the Baldwin effect briefly, explains our evolutionary model and discusses how this effect was observed in the evolutionary experiments.

2 Background

The Baldwin effect explains interactions between learning and evolution by paying attention to balances between benefit and cost of learning. The Baldwin effect consists of the following two steps (Turney, Whitley and Anderson [5]): In the first step, lifetime learning (phenotypic plasticity) gives individual agents chances to change their phenotypes. If the learned traits are useful for agents and make their fitness increase, they will spread in the next population. In the second step, if the environment is sufficiently stable, the evolutionary path finds innate traits that can replace learned traits, because of the cost of learning.

thin line: degree of dependence on learning
broken line: fitness
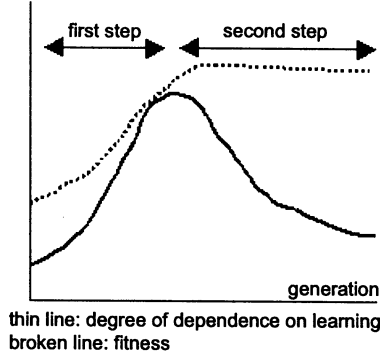
Figure 1: Two steps of the Baldwin Effect.

Through these steps, learning can accelerate the genetic acquisition of learned traits without the Lamarckian mechanism in general. Figure 1 shows the concept of the Baldwin effect roughly which consists of two steps described above.

We have adopted the iterated Prisoner's Dilemma (IPD) as a dynamic environment, which represents an abstraction of the situations causing social dilemma. Each "game" is carried out as follows:

- Two players independently choose actions from Cooperate (C) or Defect (D) without knowing the other's choice. Each player gets the score according to the payoff matrix (Table 1). We term this procedure "round".

- Above round is executed repeatedly and players compete for higher average scores.

In the only one round game, the payoff matrix makes defecting be the only dominant strategy regardless of opponent's action, and defect-defect action pair is the only Nash equilibrium. But this equilibrium is not Pareto optimal because the score of each player of cooperate-cooperate action pair is higher, which causes a dilemma. Furthermore, when the round is repeated, cooperating each other will turn out advantageous to both of them in the long run.

Table 1: Payoff matrix of Prisoner's Dilemma.

player \ opponent	cooperate	defect
cooperate	(R:3, R:3)	(S:0, T:5)
defect	(T:5, S:0)	(P:1, P:1)

(player's score, opponent's score)
$$T > R > P > S, 2R > T + S$$

3 The Model

3.1 Expression of strategies

The strategies of agents are expressed by two types of genes: genes for strategy (GS) and genes for learning (GL). GS describes deterministic strategies of IPD like Lindgren's model [6], which defines next action according to the history of actions. GL expresses whether each corresponding bit of GS is plastic or not.

A strategy of memory m has an action history h_m which is a m-length binary string as follows:

$$h_m = (a_{m-1}, \ldots, a_1, a_0)_2, \qquad (1)$$

where a_0 is the opponent's previous action ("0" represents defection and "1" represents cooperation), a_1 is the previous player's action, a_2 is the opponent's next to previous action, and so on.

GS for a strategy of memory m can be expressed by associating an action A_k (0 or 1) with each history k as follows:

$$GS = [A_0 A_1 \cdots A_{n-1}] \quad (n = 2^m). \qquad (2)$$

In GL, L_x specifies whether each phenotype of A_x is plastic (1) or not (0). Thus, GL can be expressed as follows:

$$GL = [L_0 L_1 \cdots L_{n-1}]. \qquad (3)$$

For example, the famous Tit for Tat strategy (cooperates on the first round, whatever its opponent did on the previous round) can be described by memory 2 as GS=[0101], GL=[0000].

3.2 Meta-Pavlov learning

A plastic phenotype can be changed by learning during game. We have adopted a simple learning rule termed "Meta-Pavlov" and each agent changes phenotypes according to the result of each round by referring to the Meta-Pavlov learning matrix (Table 2). It doesn't express the strategy itself but the way to change own strategy (phenotype) according to the result of the current round, though this matrix is the same as that of Pavlov strategy (Nowak and Sigmund [7]). The learning process is described as follows:

- At the beginning of the game, each agent has the same phenotype as GS itself.

- If the phenotype used in the round is plastic (the bit of GL corresponding to the phenotype is 1), the phenotype is changed to the corresponding value in the Meta-Pavlov learning matrix based on the result of the round.

Table 2: Meta-Pavlov learning matrix.

player \ opponent	cooperate	defect
cooperate	C	D
defect	D	C

- The agent uses the new strategy specified by the changed phenotype from next round on.

Take a strategy of memory 2 expressed by GS=[0001] and GL=[0011], for example of learning (Figure 2). Each phenotype represents the next action corresponding to the history of the previous round, and the underlined phenotypes are plastic.

Let us suppose that the history of previous round was "CC (player's action: cooperation, opponent's action: cooperation)" and the opponent defects at the present round. This strategy cooperates according to the phenotype and the result of the present round is "CD". The strategy changes own phenotype based on the Meta-Pavlov learning matrix according to the result of the present round, because the phenotype applied at this round is plastic. The phenotype "C" corresponding to the history "CC" is changed to "D" in this example. Therefore, this strategy chooses defection when it has the history "CC" at the next time.

The values of GS that are plastic act merely as the initial values of phenotype. Thus we represent strategies by GS with plastic genes replaced by "x" (e.g. GS=[1000], GL=[1001] → [x00x]).

3.3 Evolution

Each bit of gene is set randomly in the initial population. The round robin tournament is conducted between individuals with the strategies as are described in the previous section, under the condition in which the performed action can be changed by the noise with probability p_n. Each plastic phenotype is reset to the
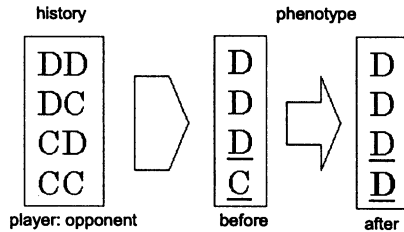
Figure 2: Example of Meta-Pavlov learning.

corresponding value of GS at the beginning of games. Rounds are repeated with the probability p_f, which is decided at the end of each round. The tournament is "ecological": The total score of each agent is regarded as a fitness value, new population is generated by the "roulette wheel selection" according to the scores, and mutation is performed on a bit-by-bit basis with probability p_m.

Average scores during the first 20 IPD games between new pair are stored, and will be used as the results of the games instead of doing games actually, so as to reduce the amount of computation. Stored scores are cleared and computed again by doing games every 500 generation.

4 Preliminary experiments

Strategies of memory 2 were investigated in the preliminary experiments. We conducted an evolutionary experiment for 2000 generations using the following parameters: $population = 1000$, $p_m = 1/1500$, $p_n = 1/25$ and $p_f = 99/100$.

The results are shown in Figure 3 and 4. The horizontal axis represents the generations. The vertical axis represents the distribution of strategies. At the same time, it also represents both "plasticity of population" (in black line) which is the ratio of "1" in all GLs and the average score (in white line). Plasticity of population is supposed to correspond to the "degree of dependence on learning" in Figure 1. The average score represents the degree of cooperation in the population, and it takes 3.0 as the maximum value when all rounds are "CC".

The evolutionary phenomena observed in a typical experiment are summarized as follows. Defective strategies ([0000], [000x] and so on) spread and made the average score decrease until about 60th generation, because these strategies can't cooperate each other. Simultaneously, partially plastic strategies ([0x0x], [00xx], [0xxx]) occupied the population. Next, around the 250th generation, more plastic strategies ([xxxx], [x0xx] and so on) established cooperation quickly, which made the plasticity and average fitness increase sharply. This transition is regarded as the first step of the Baldwin effect.

Subsequently, the plasticity of population decreased and then converged to 0.5 while keeping the average score high. Finally, the strategy [x00x] occupied the most of the population. The reason seems to be that the strategy has the necessary and adequate amount of plasticity to maintain cooperative relationships and prevent other strategies from invading in the population. This transition is regarded as the second step of

proportions of strategies occupied each population / the plasticity of population average score
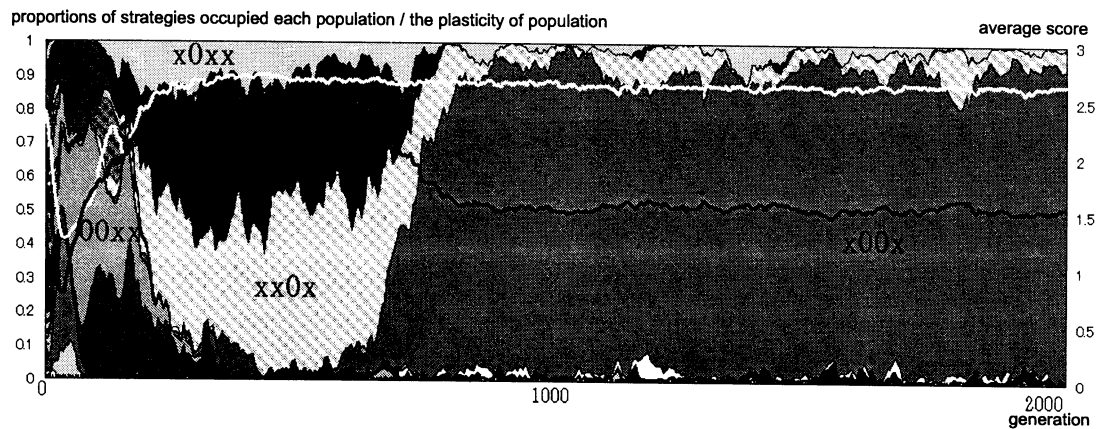
Figure 3: Experimental result (2000 generations).

proportions of strategies occupied each population
the plasticity of population average score
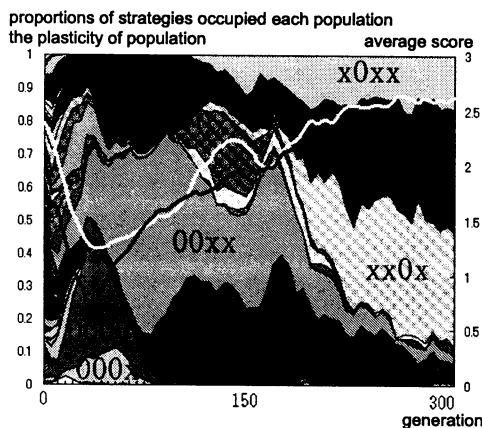
Figure 4: Experimental result (300 generations).

the Baldwin effect.

The population converged to the strategy [x00x] in all experiments, and the evolutionary phenomena described above was observed in 70% of experiments. Further analysis has shown that this strategy satisfies ESS condition in memory 2 population.

5 Conclusion

We have discussed how learning can affect the course of evolution in dynamic environments based on the results of the computational experiments on the evolution of game strategies. When we introduced the Meta-Pavlov learning to strategies as a phenotypic plasticity, population evolved to be cooperative and stable through two steps of the Baldwin effect, and

was always occupied by the strategy with a modest amount of plasticity at last.

This model could be extended in a lot of directions. One obvious direction would be to attempt to apply the automatic mechanism of blending learning with evolution in the field of multi-agent systems.

References

[1] Baldwin J. M. , "A New Factor in Evolution, " *American Naturalist*, Vol. 30, pp. 441–451, 1896.

[2] Hinton G. E. and Nowlan S. J. , "How Learning Can Guide Evolution, " *Complex Systems*, Vol. 1, pp. 495–502, 1987.

[3] Anderson R. W. , "Learning and Evolution: A Quantitative Genetics Approach, " *Journal of Theoretical Biology*, Vol. 175, pp. 89–101, 1995.

[4] Sasaki T. , Tokoro M. , "Adaptation toward Changing Environments: Why Darwinian in Nature, " *Proceedings of 4th European Conference on Artificial Life*, pp. 145–153, 1997.

[5] Turney P. , Whitley D. and Anderson R.W. , "Evolution, Learning, and Instinct: 100 Years of the Baldwin Effect, " *Evolutionary Computation*, Vol. 4, No.3, pp. 4–8, 1996.

[6] Lindgren K. , "Evolutionary Phenomena in Simple Dynamics, " *Artificial Life II*, pp. 295–311 Addison-Wesley, 1991.

[7] Nowak M. A. and Sigmund K. , "A Strategy of Win-Stay, Lose-Shift that Outperforms Tit-for-Tat in the Prisoner's Dilemma Game," *Nature*, Vol. 364, No. 1, pp. 56–58, 1993.

Evolving a Classifier System Reward Policy

Tiago Sepúlveda
INESC, R. Alves Redol, 9, 6°
1000 Lisboa
Portugal.
tiago.sepulveda@inesc.pt

Mário Rui Gomes
INESC, R. Alves Redol, 9, 6°
1000 Lisboa
Portugal.
mrg@inesc.pt

Abstract. This paper sustains that the use of an artificial life approach to model a learning classifier system enables the emergence of the classifier system own reward policy. We use one of the features of the evolution process – the Baldwin effect, to guide the emergence of a coherent reward policy without having to provide any external metric of the success of the tasks being performed. To prove our claims we developed a synthetic environment modeled after a natural ecosystem, inhabited by classifier system based agents. These agents were subjected to a process of evolution under three learning regimes. It became clear that the emergence of a discrete reward policy occurred. Furthermore, the endogenous reward policy outdid the typical fitness function reward policy optimized by the same process.

Keywords: Classifier Systems, Artificial Life, Evolution and Learning

Introduction

Learning Classifier Systems (LCS) [1] are considered to be a very promising approach for the simulation and modeling of artificial agents adaptive behavior. The power of LCS relies on the independence of system design from the specific environmental knowledge.

The operational unit of a LCS is a classifier (if-then rule). A classifier relates the information that the system acquires (encoded in the condition part of the classifier) to system behavior (resulting from the action part of the classifier), trying, this way, to establish a bond between world events and performed actions. In consequence, the set constituted by all LCS classifiers can be seen as the LCS internal world model.

A LCS can be described by a functional structure, depicted in Fig. 1, composed of:

- Operational level, where classifiers are selected to generate the system behavior according the matching between world state and classifiers conditions.

- Reward level, where the behavior of the system is evaluated and corresponding credits are assigned to the responsible classifiers.
- Learning level. After the LCS has had the opportunity to iterate several times, the most successful classifiers are chosen to be the basis of a new generation of classifiers that will replace the poorer performative ones, and, hopefully, tune the current world model.
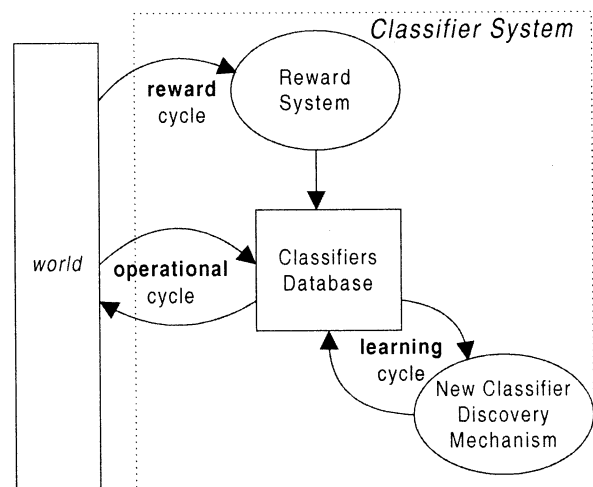
Fig. 1. Functional levels of a LCS.

This structure enables the system, not only, to rate world hypotheses (expressed in its classifiers) autonomously, but, also, to generate internally new behavioral approaches.

The Problem

LCS have been used in a broad spectrum of applications covering several fields, like robotics [2][3], user interfaces [4] or simulations [5].

One of the problems preventing a broader application of this paradigm is that the system designer is required to intervene at the reward level, providing the system a measure of its success (a reward policy). In complex environments, this turns out to be a very arduous task, highly dependent of the application domain. To take a grasp of the complexity that might be involved in the development of a good reward policy, we can look Sanza's virtual soccer game [6]. In his work, the reward policy of a LCS

based virtual soccer player was implemented by a weighted sum of several fitness functions, each related with a particular goal of a soccer game. As we might imagine, the design of such a reward system is not a trivial task.

We believe that under an evolution process the reward system can emerge by itself. The basis of our assumption is a property of natural evolution called the Baldwin effect [7]. The Baldwin effect states that since nature will favor the entities best able to survive, then the structures that acquire (learn) and use information more efficiently will be the ones that will, eventually, be dominant in a population. Stated in a more direct way, the Baldwin effect tells us that the ability to learn helps survival. Therefore, our rationale was to use an Artificial Life (AL) approach and a simulated evolution process to induce the emergence of a LCS reward policy.

Related Work

Independently of the mechanism by which reward is assigned to each classifier, it has been noted that the design of the reward system itself prevents a broader application of LCS. This is due because, as the LCS environment complexity increases, the system designer must be able to relate ever more information in order to develop an efficient reward policy.

In 1994, Dorigo and Colombetti [2] were the first to arise, explicitly, the issue of LCS reward policies. In their paper, they stand that the (standard) reward-the-result policy is not always the best solution to generate LCS environment reward. That solution implies the design of a fitness function that is not always easy to devise. Additionally, they pointed out that, in a dynamic environment, a pre-defined fitness function often did not performed well or did not meet the experimenters needs. To solve this problem, they introduced the role of the system trainer, a metaphor of a learning external supervising entity. They corroborated their claims by performing a series of experiments in a real-environment, where a mouse-sized robot equipped with two light sensors should learn to follow or avoid a light source. This solution, however, implied the existence of some supra-environmental entity (human or synthetic) to provide classifiers reward.

Using a very distinct approach, Grefenstette [8] tried also to ease the load of the LCS designer regarding reward policy. Inspired in macro-evolution processes, he proposed the replacement of the reinforcement learning based LCS by a new kind of classifier system, where classifiers sets, as a whole, were tested and bred, without having to establish any metric of the success of a particular action. This approach (named after the school where it was developed – the Pitt approach) already suggested that

evolution could play an important role in learning. Nevertheless, Grefenstette still defined a hand-designed performance measure for its classifier sets, keeping the necessity of providing the system external information.

The few approaches taken did not yet fully address the question of how LCS could develop endogenously their own reward criteria.

Framework

To conduct the experiments aimed to prove that it is possible to make emerge a reward policy through the Baldwin effect, we developed a 3D graphic environment – 3D *Saavana* (Fig. 2), modeled after a simplified natural ecosystem.
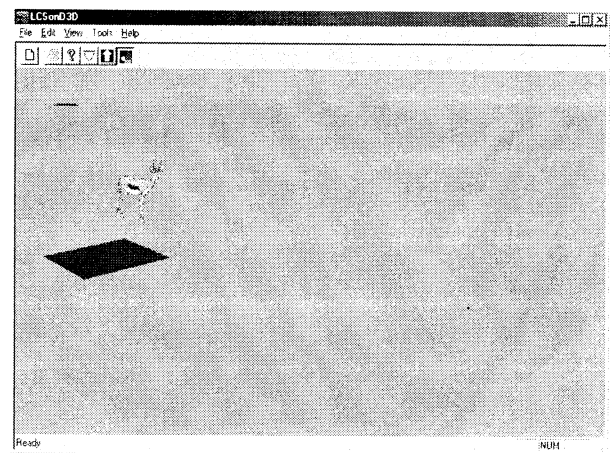
Fig 2. 3D *Saavana*.

3D *Saavana* is a *woods*-type environment [9], meaning that the environment's topology is a 2D grid, corresponding each grid cell to a possible agent location. In each location, the agent has available food resources. There is only one distinguishable type of location: grass. A grass cell is characterized by having a high food resource level. All other cells are neutral regarding food resources. The environment is inhabited by two species of agents:

* Lions, non-adaptive agents that pursue antelopes. They take energy from the antelopes when they are in the same location.
* Antelopes, LCS based adaptive agents. They collect the resources available at each location. They can breed asexually and they die if their energy is exhausted or when they reach a fixed age.

The antelopes' "biologic" features establish the conditions for an evolution process to occur [10]. There is a mechanism for increasing biological diversity – mutation, acting through breeding, and a mechanism responsible for decreasing biologic

diversity – natural selection[1]. It is under the evolution process (through the Baldwin Effect) thus established that we intend to induce the adaptation of the antelopes' LCS based control architecture in order to emerge a LCS reward policy.

Agent Control Architecture

The antelope control architecture is implemented by a hierarchical structure of LCS (Fig. 3) inspired by [3]. The core of the architecture is the classifier system that coordinates the behavior of the antelope. This module (the coordinator) chooses either the eating grass behavior or the escaping lion behavior.
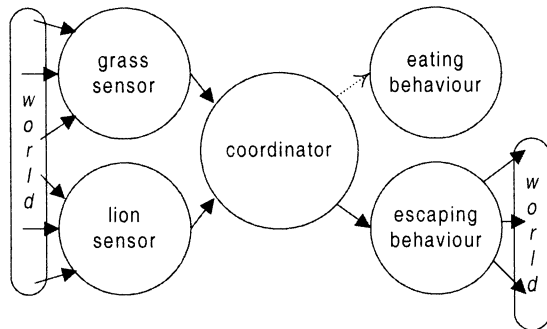
Fig. 3. Antelope control architecture.

All architecture LCS, with the exception of the coordinator, perform just the operational cycle using hand designed classifiers sets tuned, case by case, to assure an optimal LCS performance.

Coordinator

The coordinator follows the usual cycles of a LCS: operational, reward and learning cycles. The coordination module receives, from the agent (grass and lion) sensors, a distance bit indicating the proximity of the respective targets. The coordinator classifier action bit determines the module that will become active.

An example of a coordination classifier could be the classifier 00:1. The classifier condition means that there are both a lion and a grass cell in the antelope's vicinity. The classifier action indicates that the escaping module should take control of the antelope's actions (the informational flow of this example can be followed by the solid arrows of Fig. 3).

Experiments

As we have stated before, in order to simplify the learning experiments, we only allowed the adaptation

of the coordination module strategy[2]. For this reason, the overall performance of the antelopes depends on the quality of the coordination strategy present. Given the characteristics of the environment, a good coordination strategy should choose the escape behavior whenever a lion is present and the eating behavior otherwise. Using these guidelines, we rated all possible coordination classifiers (by a value expressing their *correctness*) providing the basis of a coordination strategy metric. This metric, the *coordination strategy quality* (CSQ), is defined as the sum over the coordination classifier set of the classifier correctness.

The results presented in Fig. 4 were obtained under the following circumstances:

- Grid size of 40x40.
- One lion.
- Initial antelope population, three individuals.
- 4 grass cells were generated every 10 days.

The antelopes were tested under 3 learning regimes:

1. Standard learning (SL), the coordinator learns by means of a handcrafted fitness function.
2. Parameter Learning (PL), the coordinator fitness function parameters are subject to evolution.
3. Evolving Learning (EL), the coordinator reward policy itself is subject to evolution. The reward policy is implemented as an autonomous Pitt approach LCS. Its classifiers propose numerical rewards given the variation of antelope's internal energy.

At equilibrium, antelope population oscillated between 6 and 11 individuals.
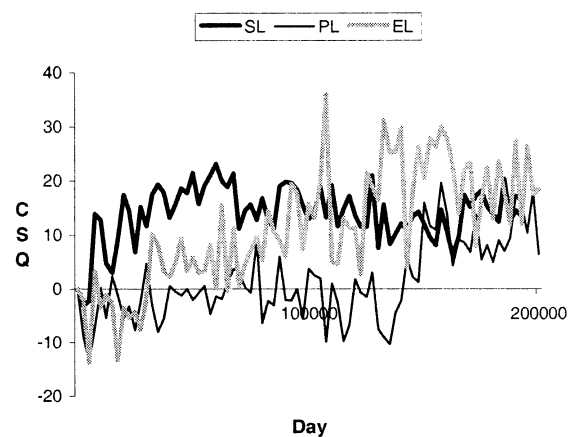
Fig. 4. Average population coordination strategy quality under standard, parameter and evolving learning regimes.

---

[1] Defined as differential reproductive success.

[2] Defined as the LCS classifier set.

We see that agents under EL regime find a good reward policy sooner than agents under other regimes, and that, after convergence, EL agents are the ones that present better reward policies. This might seem paradoxical, because in both, PL and SL, agents we introduced external, presumably better, knowledge: a hand-designed fitness function. EL agents, on the other hand, must find their own way on the environment fitness landscape. In our opinion, the main reasons that stand behind these results are:

- Poor fitness function design. Even in such a simple environment the fitness function does not take into account all pertinent information.
- Evolution is able to maintain several partial solutions through out the time, while fitness functions, due to its continuos nature, have a global effect on the reward policy.

Conclusions

The results gathered confirmed our assumption that the Baldwin effect enables the evolution of a LCS reward policy in an AL environment, and that the reward policy can be derived without giving the system any specific knowledge about the environment.

We feel it is important to explore further the interaction between evolution and learning, and that LCS can play an important role in pursuing this goal.

Furthermore, we hope in the future to use these results to establish the foundations of a LCS based evolutive hierarchical architecture, since we now have proven that, under certain conditions, a classifier system can be tuned endogenously.

Acknowledgments

References

[1] Holland, J., *Adaptation in Natural and Artificial Systems*, MIT Press, 1992.

[2] Dorigo, M. and Colombetti, M., "The Role of the Trainer in Reinforcement Learning" *Proceedings of MLC-COLT '94 Workshop on Robot Learning*, pp. 37-45, 1994.

[3] Dorigo, M. and Schnepf, U., "Genetics-based Machine Learning and Behavior Based Robotics: a New Synthesis", *IEEE Transactions on Systems, Man and Cybernetics*, 23, 1, 1993.

[4] Sanza, C., Destruel, C., Duthen, Y., "A learning method for adaptation and evolution in virtual environments", 3IA'98, 3rd International Conference on Computer Graphics and Artificial Intelligence, France, 1998.

[5] Smith, R. E., Dike, B. A., Ravichandran, B., El-Fallah, A. and Mehra, R. K., "The Fighter Aircraft LCS: A case of Different LCS Goals and Techniques", Proceedings of 2nd International Workshop on Learning Classifier Systems, USA, 1999.

[6] Sanza, C., Destruel, C. and Duthen, Y., "Learning in Real-time Environment Based on Classifier Systems", *Proceedings of the 7th International conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media'99*, 1999.

[7] Baldwin, J. M., "A New Factor in Evolution", American Naturalist, 30, 441-451, 1896.

[8] Grefenstette, J., J., "The Evolution of Strategies for Multi-agent Environments", *Adaptive Behavior 1(1)*, 65-90, 1992.

[9] Wilson, S. W., "ZCS: A Zeroth Level Classifier System", *Evolutionary Computation*, Vol. 2, 1, pp. 1-18, 1994.

[10] Colby, C., "Introduction to Evolutionary Biology", http://www.talkorigins.org/faqs/faq-intro-to-biology.html.

EXTENDING PREDICTION TERM OF GP-BASED TIME SERIES MODEL

I.Yoshihara*,   M.Numata**,   T.Aoyama*,   M.Yasunaga***,   K. Abe**

* Faculty of Engineering, Miyazaki University: 1-1 Gakuen-Kibanadai-Nishi, Miyazaki, 889-2192, Japan
(E-mail:*yoshiha@cs.miyazaki-u.ac.jp* ;Tel:+81-985-58-7384; Fax: +81-985-58-7421)
**: Graduate School of Engineering, Tohoku University: Aramaki, Aoba-ku, Sendai, 980-8579, Japan
(E-mail:*numata@largesys.ecei.tohoku.ac.jp*; Tel:+81-22-217-7074; Fax:+81-22-263-9290)
***Institute of Information Sciences and Electronics, University of Tsukuba:
1-1-1,Tennoudai, Tsukuba, 305-8573,Japan
(E-mail:*yasunaga@is.tsukuba.ac.jp*; Tel:+81-298-53-5323; Fax:+81-298-53-5206)

Abstract

A model building method of one-step-ahead prediction for time series based on genetic programming has been investigated. The method was applied to various kinds of time series e.g. computer-generated chaos, natural phenomena, financial market indices etc. Omnibus numerical experiments lead us to the conclusion that the proposed method is superior to the conventional methods.

In applying it to seismic ground motion and gold price, there appeared not a few discussion and requirement to extend the term of prediction. The longer the term of prediction, the more useful the method. This paper attempts to extend the prediction term from one-step-ahead to multi-step-ahead and to apply it to some concrete problems.

key words: *time series, one-step-ahead prediction, genetic programming, automatically define functions*

1. Introduction

Modeling and prediction of time dependent processes are important in various fields of engineering, industry, business and so forth. For prediction we need to think out a prediction model that describes essential characteristics of the system, because data do not give us any knowledge nor law by themselves but only crude information.

Most of the conventional methods essentially optimize parameters of a mathematical model that analysts or researchers provide beforehand. But because it is not easy for them to provide an appropriate math-model, an automatic construction method of one-step-ahead prediction models with genetic programming (GP) has developed [1,2,9,10].

The construction of models consists of two stages [2,9]. The first stage generates only functional form of the model that is inherited from the parent models through crossover operation of GP. The second stage optimizes parameters involved in the generated model based on back-propagation-like numerical method. In this process GP does not require giving any model, nor any special knowledge. This is a reason why we use GP. The method

has been successfully applied to many a real world problems [2,3,5,6,7,8,9].

In this paper we attempt to extend the prediction term from one-step-ahead to multi-step-ahead, because in some application we heard not a few discussion and requirement to extend the term of prediction. The possibility to extend the term of prediction must depend on nature of the system, for example, if it varies rapidly, has wide dynamic range, involves much noise and so on. We will present some concrete examples, but we have to recognize that extending prediction term is still a state-of-art technology.

Chapter 2 shows prediction model and how to build it. Chapter 3 shows application of extended model to computer-generated chaos and seismic ground motion. Chapter 4 concludes and makes a mention of future works.

2. GP-based Time Series Model

2.1 One-step-ahead prediction

One-step-ahead prediction is made using a set of finite number of past values measured from the system (Fig.1) [4].

$$\tilde{x}_t = f(x_{t-1}, x_{t-2}, \ldots, x_{t-n}) \qquad (1)$$

The prediction is to map points in delay space spanned by (x_{t-1}, x_{t-2},..., x_{t-n}) to an estimate of the future value.
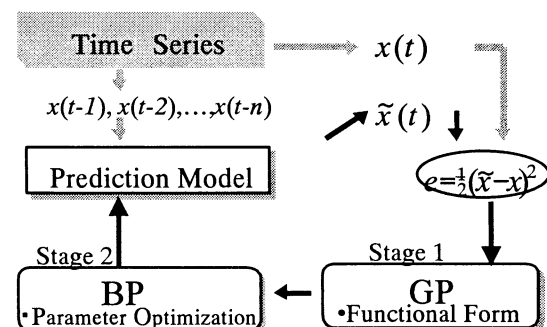
Fig.1 Procedure to build models

Most of the conventional method, such as ARMA models, is linear combination of past values. Though those models are tuned by selecting lag terms and balancing weights of the terms, it is obvious that complexity of the system cannot be completely embeded in linear models. In general, $f(\ )$ is a complicated heterogeneous functions of variables $x_{t-1}, x_{t-2}, \ldots, x_{t-n}$.

Assuming $f(\ )$ can be approximately represented by composition of arithmetic operators and elementary functions, the functional form of prediction model is described as a tree (Fig.2). The model represented by tree is embedded in structured chromosome of GP. The root of the tree is a model function and inner nodes of the tree(branching points) correspond to *node functions*, such as $+$, $-$, $\times$, $\div$, sin, cos, log, sigm(sigmoid function), abs, exp, unit, $\tan^{-1}$, max, min, mid, if, and nop (no operation)[2]. Outer nodes of the tree (leaves) correspond to input variables $x_{t-1}, x_{t-2}, \ldots, x_{t-n}$.

$$f(x)$$
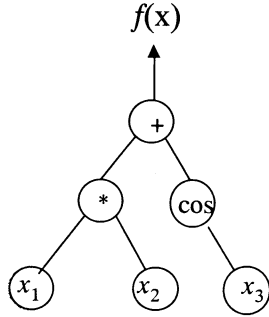
Fig.2 Model Representation

GP performs the following five steps in each generation, four of which are genetic operations and one is convergence judgment.

<initialization>

    step-0   originating population

<repetition>

    step-1   crossover
    step-2   fitness evaluation
    step-3   natural selection
    step-4   mutation
    step-5   termination

After crossover composes a functional form of the model, model parameters are optimized in according with a repetition algorithm imitating back propagation algorithm of neural network [11].

Repetition terminates, if fitness of the elite reaches a certain value given in advance, or times of repetition reaches the predefined value. In either case the elite of the last generation is regarded as the most appropriate model. GP produces and selects candidate models through "generation and test" process.

2.2 Multi-step-ahead prediction

Multi-step-ahead prediction is also given by a set of finite number of past values measured from the system.

A simple way is to derive $\tilde{x}_{t+m}$ by recurrent use of Eq.(1), where the latest values $x_{t+m-1}, x_{t+n-2}, \ldots, x_t$ are replaced by estimated ones by Eq.(1):

$$\tilde{x}_t = f(x_{t-1}, x_{t-2}, \ldots, x_{t-n}) \tag{2}$$

$$\tilde{x}_{t+1} = f(\tilde{x}_t, x_{t-1}, x_{t-2}, \ldots, x_{t-n-1}) \tag{2'}$$

$$\tilde{x}_{t+2} = f(\tilde{x}_{t+1}, \tilde{x}_t, x_{t-1}, \ldots, x_{t-n-2}) \tag{2''}$$

$$\cdots\cdots$$

Another idea to build a multi-step-ahead prediction model is using genetic programming directly. The frame work of building a model is similar to building model of Eq.(1):

$$\tilde{x}_{t+m} = g(x_{t-1}, x_{t-2}, \ldots, x_{t-n}) \tag{3}$$

Multi-step-ahead prediction model $g(\ )$ yields a future value $\tilde{x}_{t+m}$ with only past values $x_{t-1}, x_{t-2}, \ldots, x_{t-n}$, but without the future values $x_{t+m-1}, x_{t+n-2}, \ldots, x_t$.

3. Numerical Experiments

The proposed method is evaluated through numerical experiments with two kinds of test data; computer generated chaos and seismic ground motion.

3.1 Lorenz Chaos

Computer generated chaotic time series can occur only in a nonlinear system. Lorenz model is represented by differential equation systems.

$$\frac{dx}{dt} = -\sigma(x - y)$$

$$\frac{dy}{dt} = rx - y - xz \tag{4}$$

$$\frac{dx}{dt} = xy - bz$$

Setting parameters $\sigma = 10$, $r=28$, $b=8/3$, we derived a trajectory from an initial point to final point using Runge-Kutta method. 250 data of $z(t)$ are generated as time series. The first 150 data are used for model building and the remaining 100 data for validation.

Trials of multi-step-ahead prediction according to Eq.(2), Eq.(2)',...,Eq.(2)$^{(m)}$ are carried out. Predicting term "m" are varied from 0 to 4, which correspond to one-step-ahead to five-step-ahead. Some results are illustrated in Fig.3. The left charts show predicted values by Eq.(2), *i.e.* to be obtained by recurrent use of Eq.(1) and scattered graphs in the middle column show the same results. Another scattered graphs in the right column result from Eq.(3).

The left charts of Fig.3, where black line is actual and gray line is predicted, illustrate one-step-ahead prediction succeeds in coping with peaks of the time
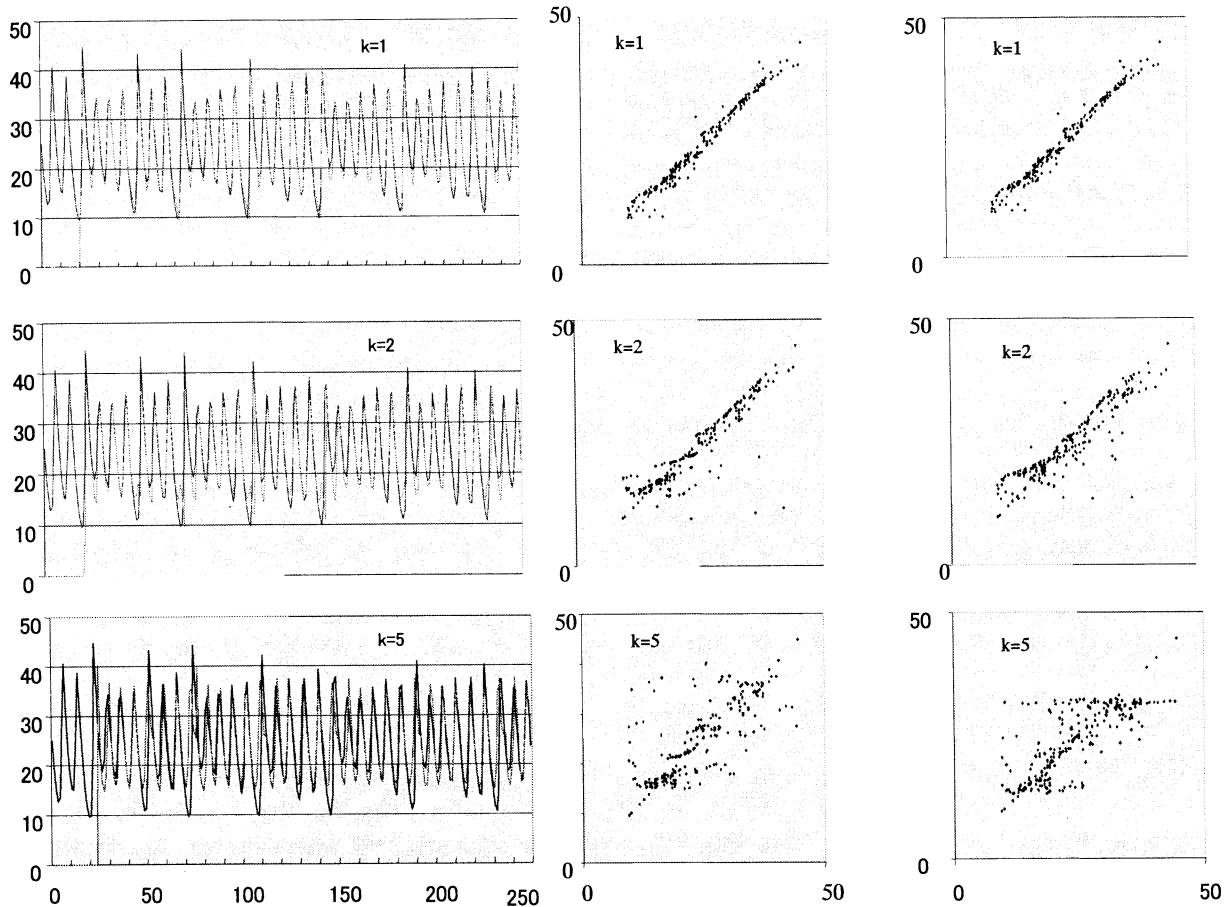
Fig.3 Prediction Results

series, but multi-step-ahead predictions are apt to miss the peaks of the time series. When "m" goes away from one, in other words, becomes two or five, forecasting errors grow rapidly. The scattered charts show prediction accuracy intuitively. If plots form a slim line, the model gives good prediction. Else if plots form a broad line or distribute widely, they involve much errors.

Fig.4 shows maximum errors of the two methods mentioned-above. The x-axis is prediction term and the y-axis is errors. Errors by recurrent use of $f(x)$ are bigger than that of $g(x)$ except m=2. GP direct method is a little better than recurrent use of one-step-ahead method, but
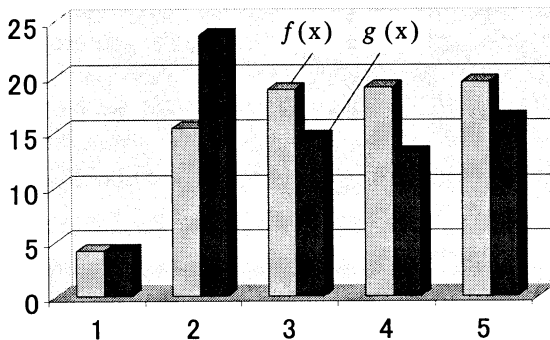
Fig.4 Comparison of errors by two models

the difference is not significantly big.

3.2 Seismic Ground Motion

Earthquakes break out frequently in the countries along the Pacific Rim. We already addressed the time series of seismic ground motion is predictable by GP-based model [10]. We again try to extend the prediction term in the same way of chaotic series.

We experiment on one of the *Hyuganada Earthquakes* "MYZ2-9NS" occurred at 14:00 on 18 March 1987, at the bottom of Hyuganada Sea that locates along the eastern coast of Kyushuu Island, Japan. A slight movement held about 12 seconds, after that, a big shock arrived and held about 10 seconds. The ground motion was measured every 10 ms. Fig.5 is one-step-ahead prediction with actual data, that involve 4096 points of data.

Let's investigate multi-step-ahead prediction based on Eq.(3) using the main part of the ground motion. Fig.6 shows actual data, one- step-ahead, five-step-ahead, and ten-step-ahead data from top to bottom respectively. The x-axis is time from 1200 to 1400 ms and the y-axis is amplitude of ground motion.

One step-ahead prediction is very close to actual values, but as prediction term becomes longer, the accuracy decreases, especially "m" is still a small number.
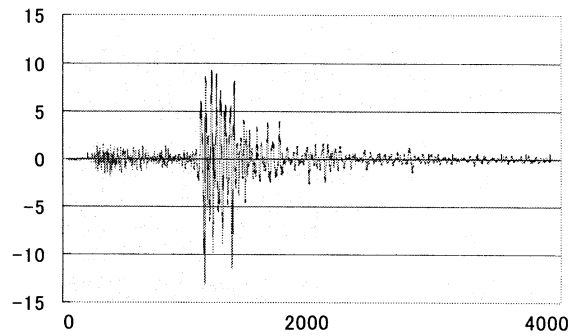
Fig.5 One-step-ahead Prediction ("MYZ2-9S")

But if "m" becomes bigger than five, the accuracy reduces moderately. Seeing the amplitude of Fig.6, we notice that the model tends to predict flatter wave than actual, as "m" increases.
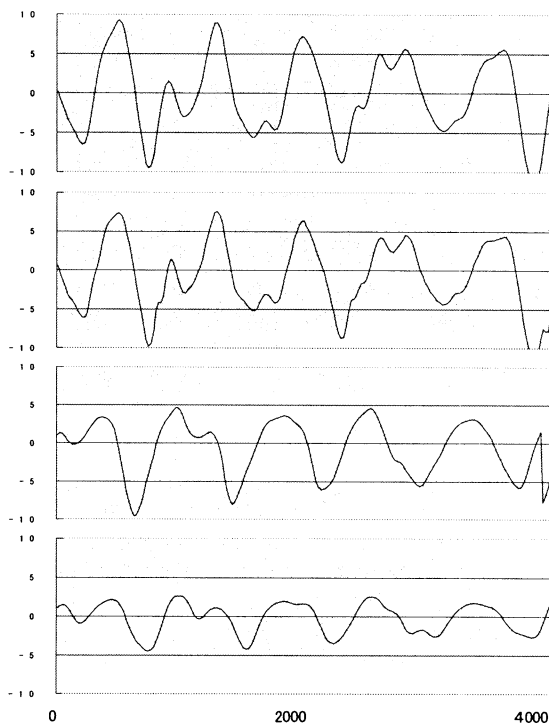
Fig.6 Multistep-ahead Prediction ("MYZ2-9S")

4 Conclusions

A model building method, that is a novel combination of standard GP and BP-like iterative algorithm, has realized one-step-ahead prediction of time series. This paper makes an attempt to extend one-step-ahead to multi-step-ahead prediction. The method was discussed through the data of computer generated chaos and seismic ground motion.

The possible terms of extended prediction depend on requirement of prediction errors. GP can generate a multi-step-ahead prediction model whose error rapidly

grows as the term of extension grows. Models generated using GP directly has a little better characteristics of prediction than that using one-step-ahead model recurrently. To extend prediction term using GP is still a state-of-the-art technology and lots of future work will be expected. More attempts will be in time on the Symposium.

Acknowledgements

References

[1] Numata,M., Sugawara,K., Yamada,S., Yoshihara,I. and Abe,K. (1998):"Time Series Prediction by Genetic Programming", Late Breaking Papers, GP-98,pp.176-179
[2] Numata,M., Sugawara,K., Yamada,S., Yoshihara,I., Abe,K.(1999),:"Time Series Prediction Modeling by Genetic Programming without Inheritance of Model Parameter",Proc. of the 4th Intnl. Symp. on Artificial Life and Robotics(AROB-IV),pp. 500-503
[3] Koza,J. 1992:"Genetic Programming", On the Programming of Computers by means of Natural Selection, MIT Press,
[4]Weingend,A.S., Greshenfeld, N.A.(ed.)(1993):"Time Series Prediction",Addison-Wesley
[5] Oakley,H.(1994):"Two Scientific Applications of Genetic Programming", Stack Filters and Non-Linear Equation Fitting to Chaotic Data, in Advances in Genetic Programming, (ed. Kenneth E.Kinnear,Jr), MIT Press,.
[6] Iba,H., deGaris,H, and Sato,T.(1995): "Recombina-tion Guidance for Numerical Genetic Programming", Proceedings of 2nd International Conference on Evolutionary Computation, IEEE Press,.
[7] Yoshihara,I., Sato,S. (1996):"Nonlinear Model Building Method with GA and GMDH", IPSJ, AI-105, p.1, (in Japanese).
[8] Kargupta,H. and Smith,R.E(1991).:"System indentification with evolving polynomial networks", Proceedings of the Fourth International Conference on Genetic Algorithms (Belew and Lashon B. Booker, ed.), San Diego, p. 370
[9] Yoshihara,I., Numata,M., Sugawara,K., Yamada,S., Abe,K.(1999): Time Series Prediction Model Building with BP-like Parameter Optimization: Proceedings of 1999 Congress on Evolutionary Computation (CEC-99), p.195
[10] Yoshihara,I., Numata,M., AoyamaI,T., Harada,T. (1999):"Prediction of Time Histories of Seismic Ground Motion using Genetic Programming", the 14th Korea Automatic Control Conf. (KACC'99), pp.E-226-E-22
[11]Rumelhart,D, Hinton,G. and illiams,R F(1986)data: "Learning representations by back-propagation errors", Nature 23, p.533.

Implementation of Real-Time Control of Assembly Line on the Back Cover of a Camera

M. H. Lee*, **K. Son***, **S. H. Han****, **J. M. Lee*****, **M. C. Lee***, **J. W. Choi***, and **Y. H. Chang******

* : School of Mechanical Engineering, Pusan National University, Pusan, KOREA
TEL) +82-51-510-2331, FAX) +82-51-512-9835, E-mail) mahlee@hyowon.cc.pusan.ac.kr
** : Dept. of Mechanical Design Engineering, Kyungnam University, Masan, KOREA
TEL) +82-551-249-2624, FAX) +82-551-243-8133, E-mail) shhan@kyungnam.ac.kr
*** : Dept. of Electronic Engineering, Pusan National University, Pusan, KOREA
TEL) +82-51-510-2378, FAX) +82-51-512-9835, E-mail) jmlee@hyowon.cc.pusan.ac.kr
**** : Graduate School of Mechanical Engineering, Pusan National University, Pusan, KOREA
TEL) +82-51-510-1456, FAX) +82-51-512-9835, E-mail) hl1lpk@hanmail.net

Abstract

This paper presents an intelligent robot control system using an off-line programming to teach precise assembly task of electronic components in a flexible way. The task investigated consists of three jobs: heat caulking test, soldering on a circuit board, and checking of soldering defects on the back cover of a camera. This study invesigates the remodelling of the most complicated cell in terms of accuracy and fault rate among the twelve cells in a camera back-cover assembly line. We have attempted to enhance soldering quality, to add task flexibility, to reduce failure rate, and to increase product reliability. This study modifies the cell structure, changes robots, improves soldering condition, implements the real-time control of assembly with vision data, and realizes an easier task teaching on off-line programming.

Key words: Back cover of camera, Automatic teaching, Off-line programming, Intelligent robot control system,

1 Introduction

Robots have been widely used for factory automation over the whole industry. The assembly automation has been generally constructed using the special-purposed robots which can hardly be applied to the other similar tasks. The generalization of an assembly cell is, therefore, one of the most challenging issues to attain a flexible manufacturing system. Automation in the assembly line of electronic components is hampered by the time delay in setting-up a task, the increased number of line stops, and the difficulty in instantaneous correction of malfunctions. In order to cope with this serious problem, we propose a method to build an assembly line using a flexible cell capable of frequent modifications. The method is particularly effective when the life cycle of product is short so that the production line must be reconstructed in a limited period. The method using the proposed cell saves the time in the rearrangement of the facility, in the line operation, and in the change of tools or components. This study focuses on the redesign of the cell, the most trouble-making one, of a total of twelve cells along the assembly line of the back cover of camera.

We developed an intelligent robot control system using an off-line programming to teach a high-precision task which involves heat caulking test, soldering on a circuit board, and fault-checking after soldering. The assembly line used has a critical drawback in the soldering unit of Cell 5, yielding a high failure rate to cause frequent stoppage and low productivity. This study replaces a four-axis SCARA robot by a six-axis one to obtain an optimum soldering position using visual information in the beginning stage of soldering. For improved soldering quality and task flexibility, a more precise soldering unit is substituted for the previous one. The scope of study covers remodelling of the cell, real-time control of assembly with a vision aid, and automatic task teaching on off-line programming. The real-time control system is completed by the installation of the redesigned mechanism of Cell 5, the calibration of an AC-2 robot, the hardware test in the assembly line, the communication between the robot and the off-line programming, the integration of controllers, and the validation field test.

2 Assembly cell

2.1 Mechanical frame

This study remodels the mechanical frame of Cell 5 to have the overall structure as shown in Fig. 1. Fig. 2 shows its components: a six-axis soldering robot(AC-2) ①, a heat-caulking test robot ②, a vision sensor ③, a stopper ④, a conveyor motor

⑤, photo and proximity sensors ⑥, an I/O controller ⑦, a control panel ⑧, a position fixture ⑨, an industrial personal computer ⑩, a pallet ⑪, a tip washer ⑫, a soldering unit ⑬, a slider for CCD camera ⑭, and a horizontal conveyor ⑮. After the heat-caulking test is passed, the process starts with the soldering job followed by the defect checking test. The frame structure is almost identical to the previous one while their moving components are replaced to handle high-precision tasks. This study searches for the optimum soldering position based on the capacity of the robot and the CCD camera installed.

2.2 Schematic diagram

The schematic diagram of Cell 5 is as shown in Fig. 3. A cell controller stores the transmitted data via RS-232C from the inspection unit with a camera, the controller of AC-2 robot, the soldering unit, and the controller of the position sensor in the heat-caulking test equipment. The stored data are sent to a personal computer of upper level via LAN.

The jobs alloted to the right, the central, and the left handling equipments are carried out in an independent
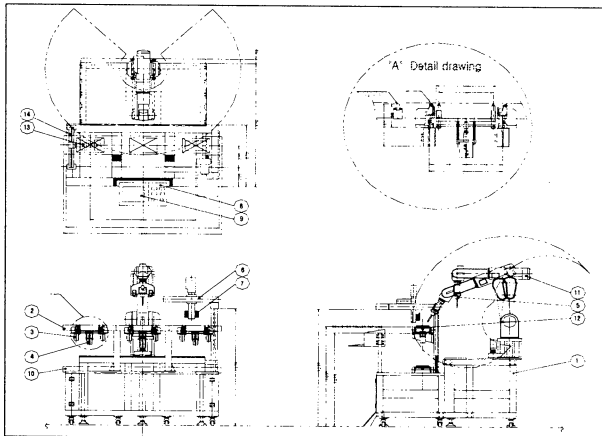
Fig. 1. Overview of the mechanical frame.
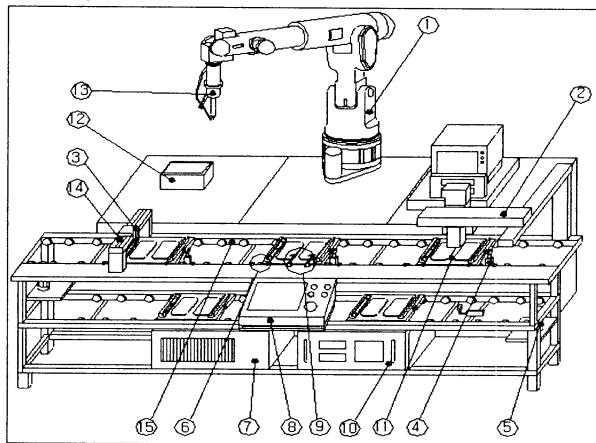
Fig. 2. Mechanical frame and components of Cell 5.
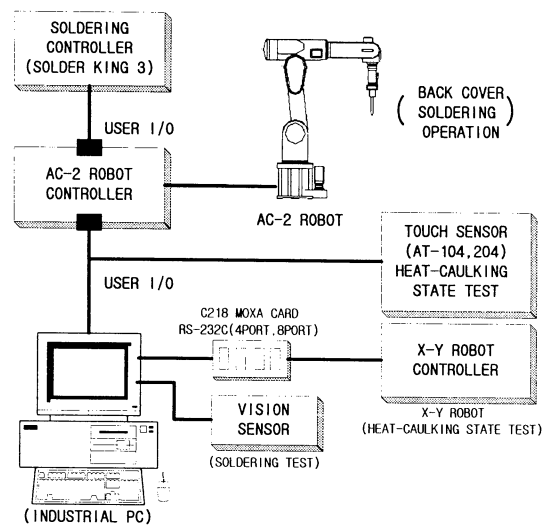
Fig. 3. Overall schematic diagram of Cell 5.

manner except when a synchronization is required to keep the predetermined tact time.

The cell controller takes charge of the flow of jobs, arbitration of the external and internal data communication, and the feedback of the data from the handling equipments. The sequential control of the three jobs is done by a DIO board on an industrial PC in Windows 95 for off-line programming and in DOS for an on-line task. Switches can be controlled with an interface to the controller via an OPTDIO board to access reset, start/stop, auto/manual, on-line/off-line, and EMG stop switches.

3 Automatic task teaching

3.1 Kinematic calibration of robots

This study intends to provide the off-line programming software with a function of automatic task teaching. The function is developed so that a soldering tip can be moved at the very spots to be soldered as programmed on a back cover of camera in a pallet. Accurate pallet positions are firstly determined in the absolute coordinates of the robot, then the locations of the other moving mechanisms are described in the relative coordinates with respect to the pallet. A visual calibration of the pallet position is necessary to successful performance of on-line tasks in the assembly line. The calibrated spatial data are feedback to the off-line programming for improved accuracy of task teachings in a more precise modeling of workspace environment. The calibration of pallet positional data is followed by the robot calibration, through which kinematic data of the robot are measured to obtain a position of the end-effector. We use a CCD camera to acquire the relative position of the end-effector to the pallet, then the positional data of pallet are transformed into the absolute robot coordinates [3, 6].
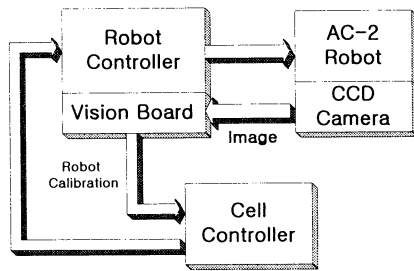
Fig. 5. Calibration of robot kinematics.
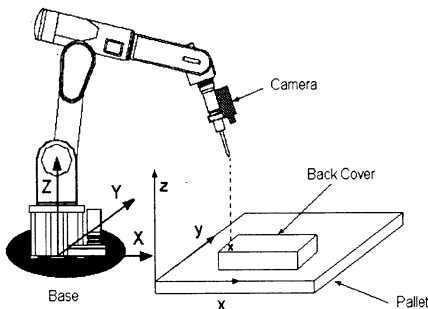
Fig. 6. Calibration of AC-2 robot with respect to pallet and end-effector.

The origin of robots is generally predetermined in robots using incremental encoders as their sensors of joint rotations because the origin plays a role of the positional reference. The origin depends on where limit sensors are installed, which are not the same in every robot. Therefore, parameters of Denavit-Hartenberg are to be accurately determined with respect to the robot origin, then a subsequent calibration is to be performed based on the positional data in both OLP and robot coordinate systems when the robot is returned to its reference position. The absolute robot coordinate system is found by this calibration. Visual data obtained from the CCD camera are used in the calculation of the relative position of the pallet to the end-effector, then the pallet position is given in the absolute coordinate system[6].

3.2 Task teaching in off-line programming

The off-line programming software[1-4] is composed of three basic units: simulation, execution, and evaluation as in Fig. 7. The simulation unit allows the user to teach a task, to plan a reference trajectory, to run dynamics and control algorithms, and finally to predict a path of the end effector. The execution unit allows one to choose a control algorithm, to load a selected algorithm, to send servo commands, and to run the robot using the real-time communication scheme. The evaluation unit gives the user three-dimensional animation of tasks, robot kinematics in either joint or space coordinate system, and comparison of the predicted and measured task results.
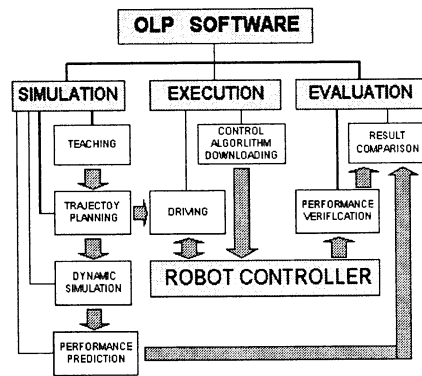
Fig. 7. Structure of off-line programming.

Task teaching and trajectory planning are followed by dynamic simulations, then a control algorithm is loaded down by a servo reset signal. Servo on and zero return commands set the robot ready for a real-time task execution. The robot carries out an on-line task by tracking the desired reference trajectory. The actual task data are fed back to the host computer, and then a task cycle is completed with the servo off command. The performance of the on-line driving can be evaluated on OLP based on the two trajectories, planned and obtained, and this calibration procedure will be repeated until the input data of OLP are properly adjusted so that the maximum difference between the two may fall within an allowance. During real-time tasks in a developed integrated system, any detection of malfunctions can issue a servo alarm signal which directs OLP to make an emergency stop of the robot.

The off-line programming is driven by menu selections and the menus to be selected are given in the upper part of the display region. OLP provides the user with seven main menus: FILE, SETUP, TEACHING, SIMULATION, EXECUTION, VIEW, and HELP. The main menus are arranged in the order of steps taken as ordinary robot tasks are processed. A previously saved file can be retrieved or a new file is made in FILE at the very beginning of the programming, then robot specifications are investigated in SETUP before the teaching starts, then a reference trajectory is planned in TEACHING. Robot dynamics and control algorithms are thoroughly analyzed to determine the best choice in SIMULATION and the selected algorithm is run on-line and the task performed is saved in EXECUTION while VIEW and HELP are offered for user's convenience.

Once one of the main menus is selected, then a list of related submenus is provided, and in a submenu the user can proceed further onto the program. Once main menu SETUP is chosen, for example, one can enter submenu specification to check detailed specifications of the robot modeled or

to change such geometric data as link dimensions, masses, and inertial properties of the robot.

In TEACHING, the user can pick up one of the trajectory planning methods listed and generate a reference path. PTP and CP methods are selectable in OLP for a wide variety of planning schemes. They include the linear interpolation, circular interpolation, cubic spline, b-spline, Bezier function, and LFPB methods. For SIMULATION Denavit-Hartenberg representation is used to describe the kinematic relationships between the adjacent robot links. The manipulator kinematics can be predicted without dynamic attributes when the end-effector does not fail to follow the desired trajectory planned in the normal task condition.

Even though the position of the camera can be represented in terms of the robot base frame, it is subjected to error. With the stable and accurate frame design, the relation between the two axes of the robot base frame and the pallet is accurate and stable always. Therefore, using the image of pallet. the position of the camera can be calibrated accurately. For this, the edges of pallet are extracted from the image first. Then using the correspondence of the lines -- a three dimensional real line and a two dimensional image line -- the translational and rotational relation between the camera and pallet axes can be obtained[5,7,8]. Following the homogeneous transformation, the accurate camera location can be obtained.

The size and shape of the back-cover can be provided from the CAD drawing data. But the location of the back-cover on the pallet might be changing during the assembly operation. To identify the precise position of the back-cover before get into the soldering operation, the points and lines on the back cover also can be utilized to identify the position. Teaching can be done either in a teaching window or in a text mode of OLP system. For operators' convenience, it can be also executed in a graphic window using a mouse and grids. Fig. 8 represents an image displayed on the OLP system, which is captured by the CCD camera. The generated task trajectory on the OLP system is converted into FARAL robot language and is sent to the robot controller through the RS232C serial communication port. Fig. 9 shows the flow diagram of automatic teaching process using the OLP system.

Fig. 9. Flowchart of automatic teaching.

4 System integration

4.1 Control system structure

The cell controller and peripherals are connected through the interfacing I/O boards and RS-232C communication boards. The soldering machine, six-axis robot, and heat-caulking test robot are equipped with controllers and are communicating with the cell controller according to the communication protocols. The pneumatic cylinder, conveyor transfer motors, conduction sensors, and position sensors are interfaced to the cell controller through the I/O boards that are plugged onto the I/O bus slots of the industrial PC. To isolate powers, the pneumatic cylinders and motors are driven by the outputs of relay switches and the position sensor data are sent to the cell controller through a photo-coupler. Solder King Model III is used for soldering operations. The visual information captured by a CCD camera and a vision board to check the soldering quality is sent to the cell controller directly.

The cell programs are modified to be applicable for the newly added peripherals. For an efficient flow of the process, a new sequence control algorithm is developed. In the algorithm, reducing the process cycles of inter-operations of peripherals under the whole system is tackled as a main goal. Fig. 10 shows the whole control structure of the soldering cell. *i.e.*, Cell 5.

Fig. 8. Relation of OLP and camera.
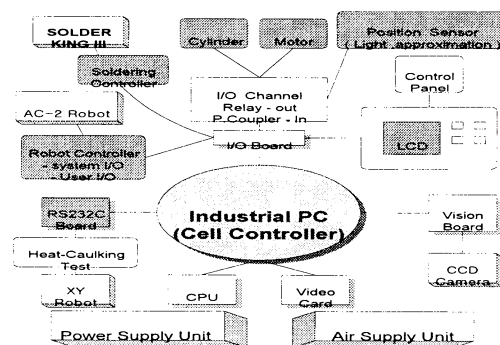
Fig. 10. Total control structure of Cell 5.

4.2 Electronic systems for the integration system

As a cell controller, a Pentium II PC is selected and operated in the Windows 95 environment. The size of RAM and VGA are increased to support OLP efficiently and an additional CRT monitor is used to replace the conventional 10.5 inch LCD display. All of the switches used for the cell control are interfaced to the cell controller with the aids of an OPTDIO board where the switches of Reset, Start/Stop, Auto/Manual, On-line/Off-line, EMG Stop and etc are located. The whole sequence of the cell operation is controlled by the industrial PC through the DIO board in the DOS environment, which is programmed by the C language. On the other hand, for the OLP system, the PC is running in the Windows 95 environment.

Even though each equipment in the cell, for examples, the soldering system, vision system, and robot system, is working independently, a process should keep a desired sequence with pre-specified tact time. This synchronization is well managed by the cell controller which supervises the total control flow, serves as a communication hub for the equipments, and acts as an OLP system. This cell controller gathers the process data from the AC-2 robot, soldering system (SOLDER KINGⅢ), soldering test equipment using machine vision, position sensor data from heat-caulking test equipment, and sends the data to the higher level assembly workstation via LAN.
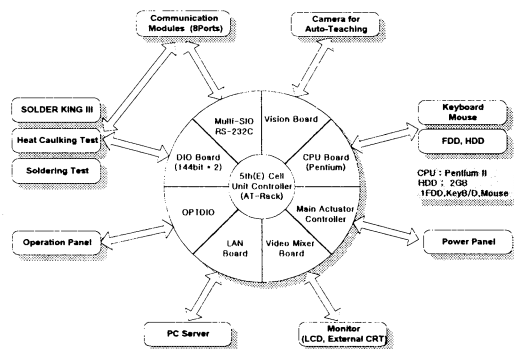
Fig. 11. Electronic system structure of Cell 5.

5 Conclusion

In this research, a working cell that is the twelve cell of a camera back-cover assembly line is selected as a test-bed for an OLP system, since the cell caused lots of defects and also required high expenses for model changes. Using the OLP system, needless to say, the model changes become easy. Also there were some of side effects, for examples, the reduction of defect rate and the increase of productivity. The major results of this research can

be listed as follows:

(1) mechanical design and implementation of Cell 5, including structural analysis,
(2) kinematic calibration and characteristic analysis of the AC-2 robot,
(3) robot calibration using camera images,
(4) interface design of the camera images to a PC in the Windows environment,
(5) auto-teaching using the off line programming system, and
(6) soldering and soldering quality inspection system.

The integrated system is under the field test at the Camera Division of Samsung Aerospace Industries, Ltd.

Acknowledgements

References

[1] K. Son, M. C. Lee, J. M. Lee, S. H. Han, M. H. Lee, and S. K. Kim, "Real-Time Evaluation of an Off-Line Programming System for SCARA Robot," Proc. of the 2nd ASCC, no. 1, pp. 89~92, 1997.

[2] Wan S. Kim, H. S. Cho, and S. K. Kim, "Robot Teaching for an Assembly Line Considering Assembly Error Distribution," '95 KACC, pp. 761~764, 1995.

[3] Y. M. Gu, K. B. Shin, and S. K. Kim, "Vision Application for Robot Application System," '94 KACC, pp. 1126~1130, 1994.

[4] C. K. Ahn, M. C. Lee, K. Son, J. M. Lee, and S. K. Kim, "Implementation of an Integrated Robot Control system for Off Line Teaching," '96 KACC, pp. 503~506, 1996.

[5] H. Freeman, "Computer Processing of Line Drawing Images," Computer Surveys, vol 6, no. 1, pp. 57-97, March, 1974.

[6] M. Y. Han and Jang M. Lee, "Precision Control of a Mobile/Task Robot Using Visual Information," Journal of the Korea Institute of Telematics and Electronics, vol. 34, no. 10, pp. 1089-1097, 1997.

[7] Anil K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall International, 1989.

[8] Yuncai Lui, Thomas S. Huang, and Olivier D. Faugeras, "Determination of Camera Location from 2-D to 3-D Line and Point Correspondences," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 12, no. 1, pp. 28-37, 1990.

Controlling Conditional Information by α-information

Ryotaro Kamimura

Information Science Laboratory, Tokai University,

1117 Kitakaname Hiratsuka Kanagawa 259-12, Japan

ryo@cc.u-tokai.ac.jp

Abstract

In this paper, we propose a conditional information controller to maximize or minimize selectively conditional information, according to the important or characteristics of input patterns. For conditional information control, we introduce α-information used to distort the ordinary Shannon information function. We have no choice but to maximize or minimize information to eliminate this distortion. Thus the distortion elimination can be used as a basic mechanism of the controller to maximize or minimize selectively information. The information controller is applied to an alphabet character recognition problem. In this problem, experimental results confirmed that conditional information is flexibly maximized or minimized, depending upon input patterns. We could also see that conditional information is a good measure to distinguish between different classes, and that the strength of conditional information is used to classify input patterns.

1 Introduction

Living systems process information from outside in very complicated ways to cope with variable surrounding conditions [1]. The systems receive huge amount of information from outside, and their principal job is to select and store information important or relevant to existence. Information should be stored as much as possible for important or relevant stimulus, while it should be as small as possible for irreverent stimulus.

Information theoretic methods have been successfully applied to neural computing. However, information, defined in multiple ways, has been simply maximized or minimized, depending upon specific problems [2], [3]. Thus, the first step to model the property of living systems is to unify information maximization and minimization methods and to develop new methods to control information, depending upon conditions and situations. In this context, we propose a condi-tional information controller, which can maximize or minimize information to be stored in a system, depending upon the importance or characteristics of input patterns. The conditional information controller maximize or minimize not information but conditional information, given input patterns. Thus, information on specific input patterns can be stored, depending upon the characteristics of input patterns.

2 Conditional Information Controller

Conditional information controllers are introduced to maximize or minimize selectively information to be stored in a system, according to the importance or characteristics of input patterns. Figure 1 shows a concept of a conditional information controller. When an input pattern is given to the controller, the controller must decide whether information on the pattern should be stored. If the pattern is critical, then information on the pattern is maximized. On the other hand, if the pattern is not so important, the information is minimized.

2.1 Information and Conditional Information

Before defining conditional information, we should explain the meaning of information used in this paper as clearly as possible, because information can be interpreted from multiple points of view. It is necessary to restrict its meaning as much as possible. We think that information is considered to be measured by the decrease in uncertainty of a system with respect to input patterns [4]. This is the information to be stored in the system. Suppose that the jth unit in a system is dependent upon the sth input pattern. Then information is difference between initial uncertainty of the
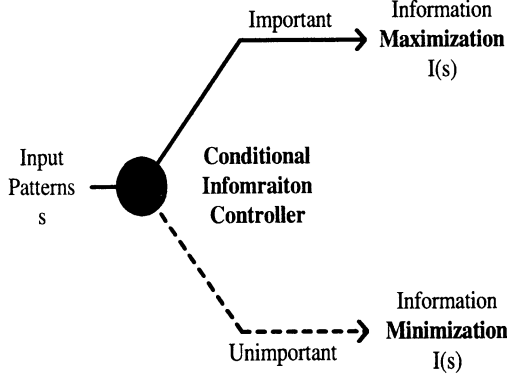
Figure 1: A conditional information controller used to maximize or minimize conditional information, depending upon input patterns.

unit and the uncertainty, given the sth input pattern:

$$I = -\sum_j p(j) \log p(j) + \sum_s \sum_j p(s)p(j \mid s) \log p(j \mid s). \quad (1)$$

Especially, when units are uniformly activated at the initial stage, that is, no information is contained in a system, information is approximated by

$$I \approx \log M + \sum_s \sum_j p(s)p(j \mid s) \log p(j \mid s), \quad (2)$$

where M is the number of units in a system, and $\log M$ is maximum uncertainty. For simplicity reason, we approximate information by this special information content to be stored in a system. This information means decrease in uncertainty by knowing input patterns.

For information specific to an input pattern, conditional information, given the sth input pattern, is needed. Conditional information for the sth input pattern is defined by

$$
\begin{aligned}
I(s) &= -\sum_j p(j) \log p(j) + \sum_j p(j \mid s) \log p(j \mid s) \\
&\approx \log M + \sum_j p(j \mid s) \log p(j \mid s) \quad (3)
\end{aligned}
$$

This information is the decrease in uncertainty only for the sth input pattern. As shown in Figure 1, if the unit is very important, the corresponding conditional information should be maximized. On the other hand, if it is not so important, the information should be as small as possible.

2.2 Conditional Information Control by α-information

Our objective is to maximize or minimize conditional information in the equation (3), depending upon different input patterns. For maximizing and minimizing conditional information for different situations, we must first introduce conditional α-information [5]:

$$I^\alpha(s) \approx \log M - \frac{1}{1-\alpha} \log \sum_j p^\alpha(j \mid s) \quad (4)$$

If we suppose that the parameter α is greater than or equal to zero, $I^\alpha(s)$ is smaller than $I(s)$ for $\alpha < 1$, and larger than $I(s)$ for $\alpha > 1$. For $\alpha \to 1$, the α-information becomes the ordinary conditional information. By the parameter α, information functions are to be distorted. Thus we call this parameter a *distortion parameter*. By combining these two measures, we can obtain the following distortion $D^\alpha(s)$:

$$D^\alpha(s) = \mid I(s) - I^\alpha(s) \mid. \quad (5)$$

This distortion is minimum, only when units are uniformly distributed (information minimization) or only one unit is turned on (information maximization). Thus, the distortion minimization corresponds to information maximization and minimization.

Especially when the distortion parameter α is zero, we have the simple conditional information

$$D^0(s) = \log M + \sum_j p(j \mid s) \log p(j \mid s). \quad (6)$$

Finally the average distortion is computed by

$$D^\alpha = \sum_s p(s)D^\alpha(s). \quad (7)$$

3 Neural Systems with Conditional Information Controller

Learning by neural networks is considered to be a process of storing information. Information can be defined as the decrease of this uncertainty about input patterns. We define information especially for hidden units, because hidden units play crucial roles in processing information in neural networks.

Intuitively the most uncertain state with respect to hidden units is a state in which all hidden units respond equally to all input patterns. On the other hand, one of the most certain states is a state in which only one hidden unit is turned on, while all other hidden units are completely off. The certain state means
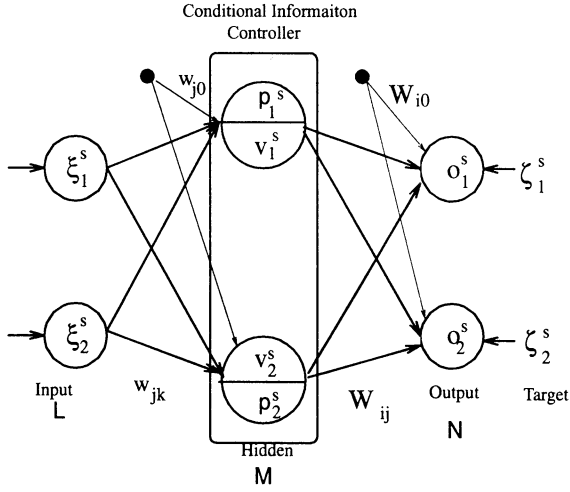
Figure 2: A neural system with a conditional information controller.

that we can definitely specify a role of a hidden unit, that is, a hidden unit turned on responds specifically to an input pattern.

Let us build a system to simulate a conditional information controller. The system is realized by using a neural network architecture shown in Figure 2. Let w_{jk} and ξ_k^s denote connections from the kth input unit to the jth hidden unit and the kth element of the sth input pattern, then total input into the jth hidden unit is computed by

$$u_j^s = \sum_k^L w_{jk}\xi_k^s,\qquad(8)$$

where L is the number of input units. The output or activity from the jth hidden unit is computed by using the ordinary sigmoidal funtion

$$v_j^s = f(u_j^s),\qquad(9)$$

where f is the ordinary sigmoidal function and defined by $f(x) = 1/(1 + \exp(-x))$.

For computing information, we use a normalized hidden unit activity

$$p_j^s = \frac{v_j^s}{\sum_m v_m^s}.\qquad(10)$$

Then the probability of the jth hidden unit, given the sth input pattern, can be approximated by

$$p(j \mid s) \approx p_j^s.\qquad(11)$$

Input patterns are supposed to be uniformly given to networks, that is,

$$p(s) \approx \frac{1}{S}.\qquad(12)$$

Thus, conditional information is approximated by

$$I(s) = \log M + \sum_j p_j^s \log p_j^s.\qquad(13)$$

The average distortion can be computed by

$$D^\alpha \approx \frac{1}{S}\sum_s \left| \frac{1}{1-\alpha}\log\sum_j (p_j^s)^\alpha + \sum_j p_j^s \log p_j^s \right|.\qquad(14)$$

By differentiating this equation with respect to input-hidden connections w_{jk}, we have update rules.

4 Application to Pattern Classification

Autoencoders with conditional information controllers were trained to reproduce exactly six alphabet letters, that is, C, G, O, H, M, N on the corresponding output units. These letters were carefully chosen to demonstrate the performance of the conditional information controller. First, these letters can be classified into two distinct groups, that is, a group, consisting of three letters: C, G, O and another group, consisting of H, M, N in terms of Hamming distance. Second in the latter group, the letter H is slightly different from the other letters in terms of Hamming distance. We aim to extract these two properties by the information controller. The number of input, hidden, and output units were 35, 10, 35 respectively for clear presentation. Even if the number of hidden units was much larger, similar experimental results were obtained by choosing appropriately the parameter α.

Figure 3 shows average information I and distortion D^α as a function of the learning parameter β, when the parameter α is 0.7. As can be seen in the figure, distortion is naturally decreased as the learning parameter β is increased. On the other hand, average conditional information is first decreased, and then the information is increased. From around $\beta = 5$, the conditional information is kept to be a constant. This suggests a possibility that conditional information is not only minimized but also maximized by choosing the approprate parameter β.

Conditional information is a good measure to classify these input patterns. In Figure 4, we can clearly
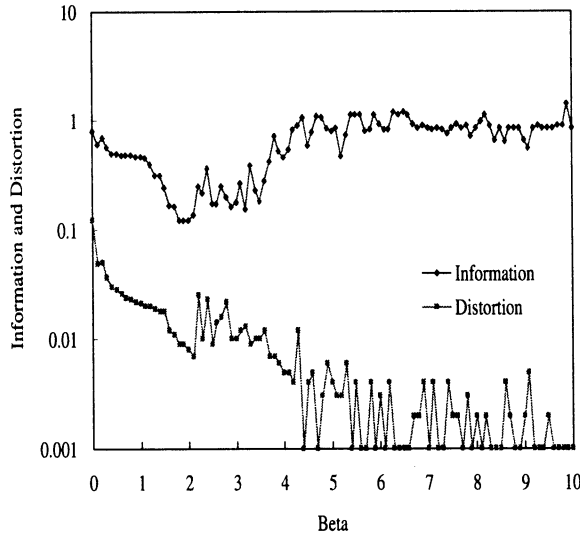
Figure 3: Information I and distortion D^α as a function of the parameter β.

Figure 4: Relative conditional information $I(s)/I$ for six different characters.

see that relative conditional information for C, G and O are smaller than that for H, M and N. We can immediately classify six characters into two groups. In addition, we can see that the conditional information for the letter H is smaller than the other two letters M, N. This means that the strength of conditional information can be used to classify input patterns.

Experimental results have confirmed that the information controller can be used to maximize or minimize conditional information by choosing appropriately the parameter α. In addition, we have seen that the information controller is good at classifying input patterns, and that the strength of conditional information is used for detailed classification.

5 Conclusion

We have proposed a conditional information controller. The controllers can be used to maximize or minimize conditional information, according to the importance of input patterns. The conditional controller has been applied to a pattern classification problem. In the problem, we have shown that the conditional controller can explicitly classify characters into appropriate groups by maximizing or minimizing conditional information.

The conditional information controller is a very general model to simulate the selectivity of living systems by using neural network architectures. Thus it can be applied to multiple phenomena, concerning complex information processing in living systems.

References

[1] S. Watanabe, *Pattern Recognition*. New York: John Wiley & Sons, 1985.

[2] G. Deco, W. Finnof, and H. G. Zimmermann, "Unsupervised mutual information criterion for elimination in supervised multilayer networks," *Neural Computation*, vol. 7, pp. 86–107, 1995.

[3] R. Kamimura, T. Takagi, and S. Nakanishi, "Improving generalization performance by information minimization," in *Proceedings of IEEE International Conference on Neural Networks*, pp. 143–148, IEEE, 1994.

[4] L. L. Gatlin, *Information Theory and Living Systems*. Columbia University Press, 1972.

[5] R. Kamimura, "Minimizing α-information for generalization and interpretation," *Algorithmica*, vol. 22, pp. 173–197, 1998.

Complexity Control Method by Stochastic Analysis for Recurrent Neural Networks

Masao Sakai
Graduate School of Eng.,
Tohoku University,
05, Aoba, Aramaki, Aoba-ku,
Sendai, 980-8579, JAPAN

Noriyasu Honma
College of Medical Sciences,
Tohoku University,
2-1, Seiryo-machi, Aoba-ku,
Sendai, 980-8575, JAPAN

Kenichi Abe
Graduate School of Eng.,
Tohoku University,
05, Aoba, Aramaki, Aoba-ku,
Sendai, 980-8579, JAPAN

(sakai,homma,abe@abe.ecei.tohoku.ac.jp)

Abstract

This paper demonstrates that the largest Lyapunov exponent λ of recurrent neural networks can be controlled by our proposed method. Our method minimizes a square error $e_\lambda = (\lambda - \lambda^{obj})^2$ by a gradient method, where λ^{obj} is desired exponent. The λ can be given as a function of the network parameters P such as connection weights and thresholds of neurons' activation. Then changes of parameters to minimize the error are given by calculating their gradients $\partial \lambda / \partial P$.

In a previous paper, we proposed a control method of λ via a direct calculation of $\partial \lambda / \partial P$ with a gradient collection through time. This method however is computationally expensive for large-scale recurrent networks and the control is unstable for recurrent networks with chaotic dynamics.

Our new method proposed in this paper is based on a stochastic relation between the complexity λ and parameters P of the network configuration under a restriction. Then the new method allows us to approximate the gradient collection in a fashion without time evolution. This approximation require only $O(N^2)$ run time while our previous method needs $O(N^5 T)$ run time for networks with N neurons and T evolution. Simulation results show that the new method can realize a "stable" control for large-scale networks with chaotic dynamics.

1 Introduction

Recurrent neural networks, consisting of units connected each other, have higher degree of parameter freedom compared with that of feedforward neural networks composed of the same number of units. Harnessing the dynamics of complicated interactions among the units, the recurrent network is expected to become a useful model for identifying and controlling the non-

linear complex dynamical systems[5]. In the case of modeling of chaotic systems from the time series, a number of authors have addressed this issue: train a network to learn the chaotic attractor[2][6]. Principle and Kuo have proposed a complexity control method which updates the weights with a forgetting function given by the largest Lyapunov exponent for feedforward networks[7]. For recurrent networks Deco and Schürman have reported that the dynamical complexity can be learned by a stochastic "sample-by-sample" update of the weights with the forgetting function[1]. There is however no direct formulation for controlling the largest Lyapunov exponent of recurrent networks.

In a previous paper, we proposed a gradient-based control method of the largest Lyapunov exponent [2]. This control method gave a control formulation explicitly, but it has several problems; the method is computationally expensive for large-scale recurrent networks and the control is unstable for recurrent networks with chaotic dynamics since a gradient collection through time diverges due to the chaotic instability.

In this paper, we propose another method in order to reduce the computational cost and realize a "stable" control for recurrent networks with chaotic dynamics. Firstly we investigate relations between the complexity and parameters of the network configuration under a restriction. The new method is based on the relation which allows us to calculate the gradient collection in a fashion without time evolution. Simulation results show that new method can control the exponent for recurrent networks with chaotic dynamics.

2 Complexity of recurrent networks

A subset of the Lyapunov exponents is used as a measure of the dynamical complexity. The complexity is defined strictly by using the complete set of the exponents. Calculation of the complete set is, howev-

er, computationally expensive. In the following, only the largest Lyapunov exponent will be concerned since it can be decided whether the systems are chaotic or not by using the largest exponent: if a system is chaotic then the largest exponent λ is greater than 0, otherwise the exponent is less than 0.

Fully connected recurrent networks composed of N units are considered. The functions of neurons are sigmoid. Letting t be discrete time, $t = 1, 2, \cdots$, the outputs of neurons x_i are governed by the following difference equations:

$$x_i(t+1) = \frac{1}{1 + \exp(-2s_i(t+1)/s_0)}, \quad (1)$$

$$s_i(t+1) = \sum_{j=1}^{N} w_{ij} x_j(t) + \theta_i, \quad (2)$$

$i = 1, 2, \cdots, N$. s_i are inputs, w_{ij} connection strengths, s_0 gain coefficients of the sigmoid functions and θ_i biases. In this case, the largest Lyapunov exponent λ is calculated by the following equations[2]

$$\lambda = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \log L(t+1). \quad (3)$$

Here $L(t)$ denotes $\|L(t)\|$ and elements of $L(t+1) = [L_1(t+1), L_2(t+1), \cdots, L_N(t+1)]^\top$ are given by

$$L_i(t+1) = \frac{2}{s_0} X_i(t+1) \sum_{j=1}^{N} w_{ij} \frac{L_j(t)}{L(t)}, \quad (4)$$

where $X_i(t) = x_i(t)(1 - x_i(t))$ and $L(0)$ is supposed by the unit vector. This implies you can calculate the largest Lyapunov exponent λ using time series of the networks, $x(0), x(1), \cdots, x(T)$ and the weights w_{ij}.

3 Control of the complexity

How do you set the Lyapunov exponent of the networks to the desired value? As mentioned in previous section, the exponent λ is a function of the network parameters such as w_{ij}. That is, the question is equivalent to how you design the parameters which generate dynamics with the desired exponent. One of the methods to achieve this design is a gradient method presented in our previous paper[2]. In the following, the gradient method is described briefly.

Letting λ^{obj} be the desired exponent, the change of parameter δw_{ij} is given by

$$\delta w_{ij} = -\eta \frac{\partial e_\lambda}{\partial w_{ij}},$$

$$= -\eta(\lambda - \lambda^{obj}) \frac{\partial \lambda}{\partial w_{ij}}, \quad (5)$$

where $e_\lambda = (\lambda - \lambda^{obj})^2/2$ is a squared error and η is a positive coefficient. If the gradient $\partial \lambda / \partial w_{ij}$ is obtained, the change of parameter δw_{ij} is able to be calculated.

Our previous control method of λ needs to calculate $\partial \lambda / \partial w_{ij}$ with a gradient collection through time. This collection through time requires numerous computational resources and it makes the control be unstable for networks with chaotic dynamics.

To solve these problems, a qualitative method based on an approximate relation between the complexity and parameters of the network configuration has been proposed previously[3]. The method is practical for large-scale networks since the method requires only $O(N^2)$ run time to control the exponent λ. However the conventional approximation method was based on experimental results rather than theoretical ground, thus the control isn't always stable.

In this section we propose a new approximation method by analyzing the relation stochastically. The main point of the following analysis is to approximate the collection through time of the length $L(t+1)$ to a fashion without time evolution since a collection through time for chaotic systems results in computational divergence due to chaotic instability. We try to get the average length L' of the $L(t)$ through time. If the L' is obtained, (3) is converted to the following:

$$\lambda = \log(L'). \quad (6)$$

Our strategy to approximate L' is to calculate the expectation of $L_i(t+1)$ in (4) as a function of a key parameter $\overline{\sigma^2}$[4]. At the first we give a restriction with respect to the network configuration which allows us to introduce the key parameter $\overline{\sigma^2}$. We initialize w_{ij} randomly, then define the biases θ_i by

$$\theta_i = -\frac{1}{2} \sum_j w_{ij}. \quad (7)$$

For large-scale networks with above configuration, a variant $\overline{\sigma^2}$ of the inputs of neurons is calculated stochastically as[3]

$$\overline{\sigma^2} = \frac{1}{12N} \sum_{i,j} w_{ij}^2. \quad (8)$$

Secondly the expectation of $\sum_j w_{ij}$, $L_j(t)/L(t)$ and X_i are calculated in order to obtain $L_i(t+1)$ as a function of $\overline{\sigma^2}$. Letting w_{ij} be defined uniformly over $[-w, w]$,

the expectation $E\left[\sum_j w_{ij}\right]$ of $\sum_j w_{ij}$ is calculated by

$$E\left[\sum_j w_{ij}\right] = \sqrt{\frac{24\overline{\sigma^2}}{\pi}}. \qquad (9)$$

At the next we suppose that all direction of the vector $\boldsymbol{L}(t)$ grow the same ratio, then we have

$$E\left[\frac{L_j(t)}{L(t)}\right] \propto \frac{1}{\sqrt{N}}. \qquad (10)$$

Lastly we calculate the expectation of $X_i(t+1)$. Considering an effect of differences between $E[X_i(t+1)]$ and $X_i(t+1)$ on the difference between (3) and (6), the expectation X_i' of $X_i(t+1)$ with respect to time is defined by

$$X_i' = X' + \Delta X_i, \qquad (11)$$

where X' is the average of X_i'. From (4), (9)$\sim$(11) the expectation of $L_i(t+1)$ given by

$$E[L_i(t+1)] \propto \frac{2}{s_0}(X' + \Delta X_i) \cdot \sqrt{\frac{24\overline{\sigma^2}}{\pi}} \cdot \frac{1}{\sqrt{N}}. \qquad (12)$$

Finally the expectation of $\boldsymbol{L}(t+1)$, $\boldsymbol{L}'$ is given by

$$\begin{aligned}
\boldsymbol{L}' &= k_1\sqrt{\frac{\overline{\sigma^2}}{s_0{}^2 N}} \cdot \sqrt{\sum_{i=1}^{N}(X' + \Delta X_i)^2}, \\
&\approx k_1\sqrt{\frac{\overline{\sigma^2}}{s_0{}^2}} \cdot (X' + k_2),
\end{aligned} \qquad (13)$$

where k_1, k_2 are positive constants.
From (13) we can calculate (6) if the expectation X' is obtained. Linearizing the activation functions of neurons in (1) partially, the $X_i(t)$ is approximated by

$$X_i(t) \approx \begin{cases} \dfrac{1}{4} - \dfrac{1}{4s_0^2}s_i^2(t), & (-s_0 < s_i(t) < s_0), \\ 0, & (otherwise). \end{cases}$$
$$(14)$$

The probability density of the expectation of the input $s_i(t)$ is defined by the following normal distribution[3].

$$g(x, 0, \overline{\sigma^2}) = \frac{1}{\sqrt{2\pi\overline{\sigma^2}}}\exp\left(-\frac{x^2}{2\overline{\sigma^2}}\right). \qquad (15)$$

The average of X_i', X' is calculated by (14), (15) as

$$X' \approx \int_{-\infty}^{\infty}\left(\frac{1}{4} - \frac{1}{4s_0^2}x^2\right) \cdot g(x, 0, \overline{\sigma^2}) \; dx,$$

$$\begin{aligned}
&= \int_{-s_0}^{s_0}\left(\frac{1}{4} - \frac{1}{4s_0^2}x^2\right) \cdot g(x, 0, \overline{\sigma^2}) \; dx, \\
&= \frac{\sqrt{\overline{\sigma^2}}}{2s_0} \cdot g\left(\frac{s_0}{\sqrt{\overline{\sigma^2}}}, 0, 1\right) \\
&\quad + \left(\frac{1}{2} - \frac{\overline{\sigma^2}}{2s_0^2}\right)\int_0^{\frac{s_0}{\sqrt{\overline{\sigma^2}}}} g(x, 0, 1)\, dx. \quad (16)
\end{aligned}$$

By the following approximation of Taylor series as

$$\exp\left(-\frac{x^2}{2}\right) \approx \begin{cases} \displaystyle\sum_{n=0}^{5}\frac{(-1)^n x^{2n}}{n! \cdot 2^n}, & (-2 < x < 2), \\ \\ 0, & (otherwise), \end{cases} \qquad (17)$$

we have

$$\int_0^{\frac{s_0}{\sqrt{\overline{\sigma^2}}}} g(x, 0, 1) \; dx$$

$$= \begin{cases} \dfrac{s_0}{\sqrt{2\pi\overline{\sigma^2}}}\displaystyle\sum_{n=0}^{5}\frac{(-1)^n}{n! \cdot 2^n(2n+1)} \cdot \left(\frac{s_0^2}{\overline{\sigma^2}}\right)^n, \\ \qquad \cdots \quad \left(\dfrac{s_0{}^2}{4} < \overline{\sigma^2} < \infty\right), \\ \dfrac{1}{2}, \quad \cdots \quad \left(0 \le \overline{\sigma^2} \le \dfrac{s_0{}^2}{4}\right). \end{cases}$$

$$(18)$$

Finally from (6), (13), (15), (16) and (18) you can have the approximate relation between λ and $\overline{\sigma^2}$.
From above results, you have

$$\frac{\partial\lambda}{\partial w_{ij}} = \frac{\partial\lambda}{\partial\overline{\sigma^2}} \cdot \frac{\partial\overline{\sigma^2}}{\partial w_{ij}}. \qquad (19)$$

$\partial\overline{\sigma^2}/\partial w_{ij}$ are calculated by (8) as

$$\frac{\partial\overline{\sigma^2}}{\partial w_{ij}} = \frac{1}{6N}w_{ij}. \qquad (20)$$

$\partial\lambda/\partial\overline{\sigma^2}$ are calculated by (6), (13) as

$$\frac{\partial\lambda}{\partial\overline{\sigma^2}} = \frac{1}{2\overline{\sigma^2}} + \frac{1}{(X' + k_2)} \cdot \frac{\partial X'}{\partial\overline{\sigma^2}}. \qquad (21)$$

$\partial X'/\partial\overline{\sigma^2}$ are calculated by (16) and (18) as

$$\frac{\partial X'}{\partial \overline{\sigma^2}} = \begin{cases} \frac{1}{4\sqrt{2\pi s_0^2 \overline{\sigma^2}}} \left\{ \left(1 + \frac{s_0^2}{\sigma^2}\right) \exp\left(-\frac{s_0^2}{2\sigma^2}\right) \right. \\ \left. + \sum_{n=0}^{5} \frac{(-1)^n s_0^{2n}}{n! \cdot 2^n (\overline{\sigma^2})^n} \left(\frac{2n-1}{2n+1} - \frac{s_0^2}{\overline{\sigma^2}}\right) \right\}, \\ \qquad \cdots \quad \left(\frac{s_0^2}{4} < \overline{\sigma^2} < \infty\right), \\ \frac{1}{4\sqrt{2\pi s_0^2 \overline{\sigma^2}}} \cdot \left(1 + \frac{s_0^2}{\sigma^2}\right) \exp\left(-\frac{s_0^2}{2\sigma^2}\right) - \frac{1}{4s_0^2}, \\ \qquad \cdots \quad \left(0 \le \overline{\sigma^2} \le \frac{s_0^2}{4}\right). \end{cases}$$

(22)

The partial differential coefficients $\partial \lambda / \partial w_{ij}$ are then calculated by substituting (20) $\sim$ (22) into (19).

4 Simulation results

Our new approximation methods have been tested on a design task which requires the fully connected networks to have a desired value of the largest Lyapunov exponent. In this task, the connection weights of the network were initialized randomly, then changed by our method. The network controlled was composed of 100 neurons and $k_1 = 3.3, k_2 = 0.04, s_0 = 0.2$ and $T = 100$.

Figure 1 shows the exponent λ is controlled to the value of more complex dynamics by our method. The desired value of the exponent λ^{obj} was 1. Note that the exponent λ converges to the desired value.

5 Conclusions

In this paper we analyzed a stochastic relation between the largest Lyapunov exponent and network parameters for large-scale fully connected recurrent networks with asymmetric connection weights w_{ij} and biases $\theta_i = -\frac{1}{2} \sum_j w_{ij}$. The relation allows us to approximate the gradient collection through time in a fashion without time evolution.

Simulation results show that effectiveness of our proposed method with respect to the computational cost and the stable control of the Lyapunov exponent of recurrent networks with chaotic dynamics.
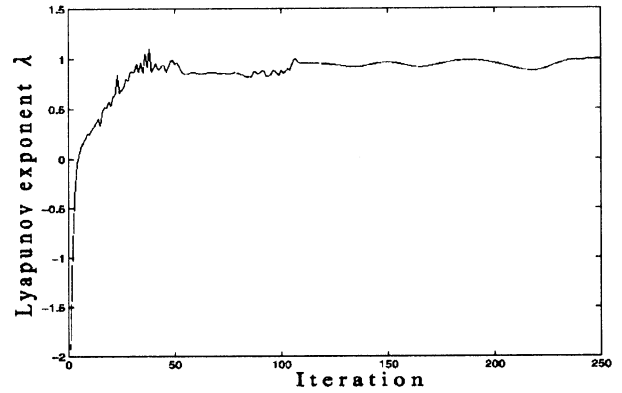
Figure 1: The Lyapunov exponent λ as a function of the iteration by approximation method ($\lambda^{obj} = 1$).

References

[1] G. Deco and B. Schürmann, "Dynamic modelling chaotic time series," *Computational Learning Theory and Neural Learning Systems*, Vol.4 of Making learning Systems Practical, MIT Press, Cambridge, Mass., 1997, chapter 9, pp.137-153

[2] N. Honma, M. Sakai, K. Abe and H. Takeda, "Complexity Control Methods of Dynamics in Recurrent Neural Networks(in Japanese)," *Trans. SICE*, Vol.35, No.1, pp.138-143.

[3] N. Honma, K. Kitagawa and K. Abe, "Effect of Complexity on Learning Ability of Recurrent Neural Networks," *Proc. of the 2nd AROB*, 1997, pp.58-61.

[4] N. Honma, K. Kitagawa, K. Abe and H. Takeda, "An Autonomous Criterion of Learning Methods for Recurrent Neural Networks," *Proc. of the 2nd ASCC*, Vol.2, 1997, pp.219-222.

[5] K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. Neural Networks*, Vol.1, No.1, 1990, pp.4-27.

[6] J. C. Principle, A. Rathie and J. Kuo, "Prediction of chaotic time series with neural networks and the issue of dynamic modeling," *International Journal of Bifurcation and Chaos*, Vol.2, No.4, 1992, pp.989-996.

[7] J. C. Principle and J. Kuo, "Dynamic modelling chaotic time series with neural networks," In G. Tesauro et al.(Eds.), Neural Information Processing System 7, 1995, pp.311-318

Control structure of stabilizing controllers for linear minimum phase single-input and single-output systems

Kou Yamada
Department of Electrical and Information Engineering
Yamagata University
4-3-16 Jonan, Yonezawa 992-8510 Japan

Abstract

The purpose of the present paper is to give control structure of stabilizing controller for the minimum phase systems and also give some characteristic of stabilizing controller for the minimum phase systems.

1 introduction

In the present paper, we examine a parametrization for the class of all proper stabilizing controllers for single-input and single-output time-invariant continuous-time minimum phase systems. The parametrization problem is the problem in which all stabilizing controllers for a plant are sought [1]-[6]. Since this parametrization can successfully search all proper stabilizing controllers, it is used as a basic tool for many control problems.

For an asymptotically stable plant, a parametrization of all stabilizing controllers for the plant has a structure identical to that of Internal Model Control [5]. This implies that Internal Model Control for the stable system is not restricted by control structure. That is, when control system for an asymptotically stable plant is designed, to adopt Internal Model Control is natural selection. For an unstable plant, the structure of a parametrization of all stabilizing controllers for the plant has full-order state feedback, including a full-order observer and any asymptotically stable rational function $Q(s)$[3]. Glaria and Goodwin gave a simple parametrization for minimum phase systems; however, one practical difficulty remains: the parametrization of the stabilizing controller $C(s)$ given by Glaria and Goodwin generally includes improper controllers. In practical application, the controller $C(s)$ is required to be proper. If a parametrization of all proper stabilizing controllers for the plant can be obtained, a parametrization of all stabilizing controllers for minimum phase systems can be applied to several applications. Yamada expanded the result described in [6] and gave an exact parametrization of all proper stabilizing controllers for the single-input and single-output continuous time minimum phase strictly proper systems [8]. But [8] did not consider the control structure of the stabilizing controllers for the minimum phase systems nevertheless it is important problem to clarify the control structure of the stabilizing controller.

The purpose of the present paper is to give control structure of stabilizing controller for the minimum phase systems and also give a characteristic of stabilizing controller for the minimum phase systems. First of all, a parametrization of all stabilizing controllers for the minimum phase systems is summarized briefly. Some control characteristics are also shown. A parametrization of all strong stabilizing controllers, that is asymptotically stable stabilizing controllers, is given. A control structure of stabilizing controllers for the minimum phase systems is clarified. This result is a fundamental part of the control design for minimum phase systems but necessarily unstable.

2 Parametrization

Let us consider a single-input and single-output continuous time system $G(s) \in R(s)$. Here, $G(s)$ is assumed to be strictly proper and of minimum phase, that is, $G(s)$ has no zeroes in the closed right half plane. In the present paper, we clarify a control structure of stabilizing controllers $C(s)$ for $G(s)$ such that the unity feedback control system given by

$$\begin{cases} y &= G(s)u \\ u &= -C(s)y \end{cases} \tag{1}$$

is internally stable.

Before clarifying the control structure of stabilizing proper controllers $C(s)$ for $G(s)$. Preliminary results of a parametrization of all proper controllers for the minimum phase systems shown in [8] are summarized. When the plant $G(s)$ is biproper, a parametrization of all controllers for the minimum phase system $G(s)$ is summarized as following theorem.

Theorem 1 *If $G(s)$ is biproper and of minimum phase, a parametrization of all proper stabilizing controllers for $G(s)$ is denoted by*

$$C(s) = \frac{1}{Q(s)} - \frac{1}{G(s)}, \qquad (2)$$

where $1/Q(s)$ is any nonsingular proper minimum phase rational transfer function.

When the stabilizing controllers (2) are used, some control characteristics are summarized as follows. The sensitivity function $S(s)$ and the complementary sensitivity function $T(s)$ are written by

$$S(s) = \frac{Q(s)}{G(s)} \qquad (3)$$

and

$$T(s) = 1 - \frac{Q(s)}{G(s)}, \qquad (4)$$

respectively. The sensitivity function and the complementary sensitivity function can be shaped very easily using free parameter $Q(s)$. The strong stabilizing controller, that is the asymptotically stable stabilizing controller, is parametrized as following theorem.

Theorem 2 *If $G(s)$ is biproper and of minimum phase, a parametrization of all proper strong stabilizing controllers for $G(s)$, that is a parametrization of all proper asymptotically stable stabilizing controllers for $G(s)$, is denoted by*

$$C(s) = \frac{1}{Q(s)} - \frac{1}{G(s)}, \qquad (5)$$

where $Q(s)$ is any unimodular rational function, that is $Q(s) \in RH_\infty$ and $1/Q(s) \in RH_\infty$.

(Proof) It was proved that a parametrization of all stabilizing controllers for the minimum phase system $G(s)$ is given in [8]. Therefore proof of this theorem is sufficient to show the stabilizing controller is asymptotically stable if and only if $Q(s)$ is unimodular.

The sufficiency will be shown. That is, if $Q(s)$ is unimodular, it will be shown that the controller $C(s)$ written by (5) is asymptotically stable. Since both $1/Q(s)$ and $1/G_m(s)$ are asymptotically stale, the controller $C(s)$ written by (5) is evidently asymptotically stable. According to Theorem 1, $Q(s)$ is requried to be asymptotically stable. Therefore $Q(s)$ is unimodular.

The necessity will be shown. That is, if the controller $C(s)$ is asymptotically stable, it will be shown that when the controller $C(s)$ is written by (5), $1/Q(s)$ is asymptotically stable. $1/Q(s)$ is rewritten by

$$\frac{1}{Q(s)} = -\frac{1}{G(s)} - C(s). \qquad (6)$$

Since both $C(s)$ and $1/G_m(s)$ are asymptotically stale, $1/Q(s)$ is asymptotically stable. According to Theorem 1, $Q(s)$ is requried to be asymptotically stable. Therefore $Q(s)$ is unimodular.

From above discussion we have thus proved this theorem. ∎

When the plant $G(s)$ is biproper, a parametrization of all controllers for the minimum phase system $G(s)$ is summarized as following theorem.

Theorem 3 *There exists $K(s)$ that satisfies the following expressions:*

- *$G(s) + K(s)$ is of minimum phase.*

- *$K(s)$ is biproper and asymptotically stable.*

Using above mentioned $K(s)$, the parametrization of all proper stabilizing controllers $C(s)$ for $G(s)$ is given by

$$C(s) = \frac{C_f(s)}{1 + C_f(s)K(s)}, \qquad (7)$$
$$\lim_{\omega \to \infty} 1 + C_f(j\omega)K(j\omega) \neq 0$$

where $C_f(s)$ is denoted by

$$C_f(s) = \frac{1}{Q(s)} - \frac{1}{G(s) + K(s)}, \qquad (8)$$

and the free parameter $1/Q(s)$ is any nonsingular proper minimum phase rational transfer function.

When the stabilizing controllers (8) are used, some control characteristics are summarized as follows. The sensitivity function $S(s)$ and the complementary sensitivity function $T(s)$ are written by

$$S(s) = \frac{G(s)Q(s) + (G(s) + K(s))K(s)}{(G(s) + K(s))^2} \qquad (9)$$

and

$$\begin{aligned} T(s) &= 1 - S(s) \\ &= \frac{G(s)\{(G(s) + K(s)) - Q(s)\}}{(G(s) + K(s))^2}, \end{aligned} \qquad (10)$$

respectively. The sensitivity function and the complementary sensitivity function can be shaped very easily using free parameter $Q(s)$.

3 Controller structure for the biproper systems

In this section, we clarify the control structure of stabilizing controllers for the minimum phase biproper system.

A simple control structure of stabilizing controllers for the minimum phase biproper system is summarized as Theorem 4.

Theorem 4 *A stabilizing controller for the linear minimum phase single-input and single-output systems has following structure.*
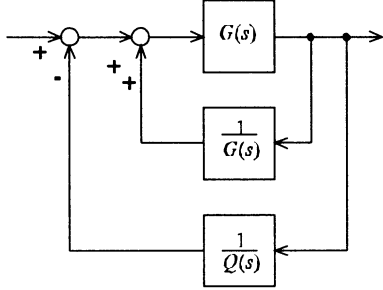
Figure 1: Simple controller structure for biproper system

(Proof) Proof of this theorem is obtained by simple similarity transformation of block diagram. Therefore details of this theorem are omitted. ∎

From Theorem 4, we have following theorem.

Theorem 5 *Using any biproper rational function $G_m(s)$, a stabilizing controller for the linear minimum phase single-input and single-output systems has following structure.*
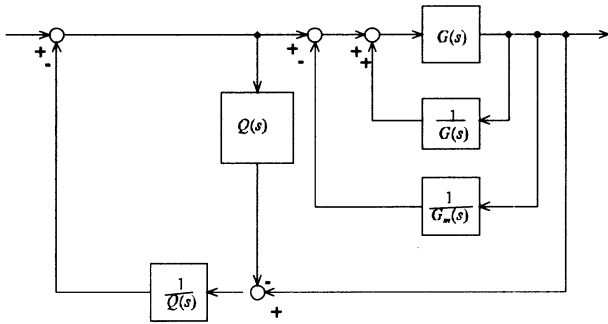
Figure 2: Simple controller structure for biproper system

(Proof) Proof of this theorem is obtained by simple similarity transformation of block diagram. Therefore details of this theorem are omitted. ∎

Remarks 1 *Since the block diagrams encircled with dotted line in Fig. 2 is equivalent to $G_m(s)$, the core structure of stabilizing control structure is rewritten by Fig. 3 .*
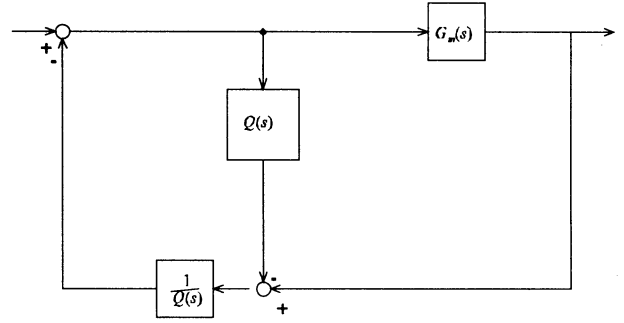
Figure 3: Simple controller structure for biproper system

This figure shows a striking feature of a stabilizing controller for the minimum phase systems. That is this structure is shown only for stabilizing controllers for the minimum phase system.

4 Controller structure for strictly proper systems

In this section, we clarify the control structure of stabilizing controllers for the minimum phase strictly proper system.

A simple control structure of stabilizing controllers for the minimum phase strictly proper system is summarized as following theorem.

Theorem 6 *A stabilizing controller for the linear minimum phase single-input and single-output systems has following structure.*
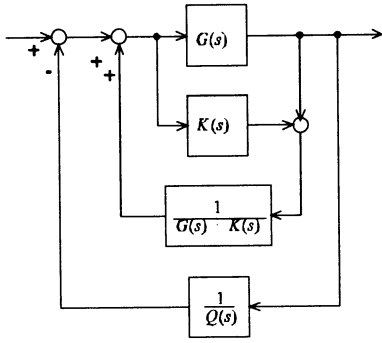
Figure 4: Simple controller structure for strictly proper systems

(Proof) Proof of this theorem is obtained by simple similarity transformation of block diagram. Therefore details of this theorem are omitted. ∎

From Theorem 6, we have following theorem.

Theorem 7 *Since the block diagrams encircled with dotted line is equivalent to $G_m(s)$, a stabilizing controller for the linear minimum phase single-input and single-output systems has following structure.*
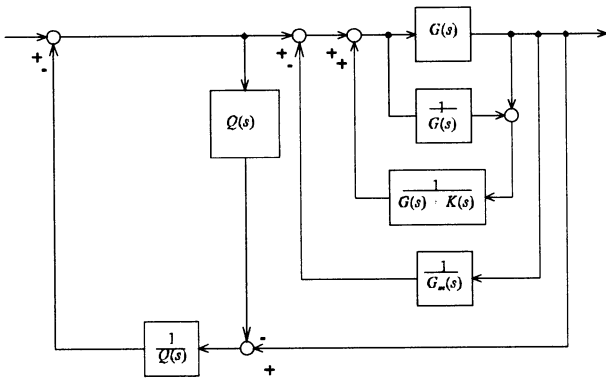
Figure 5: Simple controller structure for the strictly proper system

(Proof) Proof of this theorem is obtained by simple similarity transformation of block diagram. Therefore details of this theorem are omitted. ∎

5 Conclusion

The present paper clarified a control structure of stabilizing controller for the minimum phase sys-

tems and also gave a characteristic of stabilizing controller for the minimum phase systems and gave a parametrization of all strong stabilizing controllers, that is asymptotically stable stabilizing controller.

References

[1] G. Zames: Feedback and optimal sensitivity: model reference transformations, multiplicative seminorms and approximate inverse, IEEE Trans. on AC, AC26, 301/320(1981)

[2] D.C. Youla, H. Jabr and J.J. Bongiorno: Modern Wiener-Hopf design of optimal controllers. Part I, IEEE Trans. on AC, AC21, 3/13(1976)

[3] K. Zhou, J.C. Doyle and K. Glover: Robust and optimal control, Prentice-Hall (1996)

[4] C.A. Desoer, R.W. Liu, J. Murray and R. Saeks: Feedback system design: The fractional representation approach to analysis and synthesis, IEEE Trans. on AC, AC25,399/412(1980)

[5] M. Morari and E. Zafiriou: Robust Process Control, Prentice Hall (1989)

[6] J.J. Glaria and G.C. Goodwin: A parametrization for the class of all stabilizing controllers for linear minimum phase systems, IEEE Trans. on AC, AC39-2, 433/434(1994)

[7] M. Vidyasagar: Control System Synthesis–A factorization approach–, MIT Press (1985)

[8] K. Yamada: A parameterization of all stabilizing controllers for linear minimum phase single-input and single-output systems, SMC'99 to be published

Relation between saturation and settling time of antiwindup control using left coprime factorization method

K.Yamada
Dept. of Elec. and Inf. Eng
Yamagata University
4-3-16 Jonan, Yonezawa 992-8510 Japan

Y.Funami
Det. of Computer Controlled Mechanics
Shonai College of Industry and Technology
3-57-4 Kyoden, Sakata 998-0102 Japan

Abstract

This paper examines a anti-windup control problem. In the present paper, we consider the effectiveness of using the freedom of the left coprime factorization toward anti-windup phenomenon. At first, we will show both H_2 and H_∞ norms can be arbitrarily small using the freedom of the left coprime factorization. Next relation between saturation and settling time of antiwindup control using left coprime factorization method is considered.

1 introduction

This paper examines a anti-windup control problem. Windup phenomenon often occurs in the control systems, when the control systems have the limitation of admissible control input. In general, admissible control input is limited from various reasons when we control real systems. For example actuators for mechanical systems have a restriction to permit voltage input to save its own circuit and to have a limit of its performance. This control input saturation is seen in a lot of systems. Since real physical and mechanical systems can not permit unbounded input, the control input saturation occurs. When we design a control system with ignoring the control input saturation, it is known that not only we can not get desirable control characteristics but also the system stability dose not be guaranteed [1]-[8].

To solve this problem, several papers have been considered on the stability problem for systems having the control input saturation [1, 2]. References [1, 2] examine the stability problem for the systems with state feedback control and observer. The object of references [1, 2] is to obtain a stability condition for the control systems having a general structure. Among such design methods to be conscious of existence of the control input saturation, Internal Model Control is well known as a design method that guarantees the stability even if the control input saturation exists [3]. The fact that the stability of the system is independent of the control input saturation, is suitable for the designer. Although Doyle et al. pointed

out the problem that Internal Model Control subject to the control input saturation does not have robust servo characteristics and showed an example [8]. This means that under certain circumstances, error between the reference input and the output exists in Internal Model Control system with the control input saturation. Yamada considered the reason why error between the reference input and the output appears in Internal Model Control with the control input saturation [9]. He gave a method to design Internal Model Control having robust servo characteristics using the idea of Internal Perturbed-Model Control [9]. This resolved the problem on the robust servo characteristics of Internal Model Control and tractability of Internal Model Control is increased. In addition, Internal Model Control is well-known to have the property of anti-windup characteristics and easy to specify control characteristics. [10] showed that the reason why Internal Model Control has anti-windup characteristics. In this way, Internal Model Control is useful control design method for anti-windup phenomenon, but Internal Model Control can be applied only to asymptotically stable systems. For unstable plants, a anti-windup control design method using left coprime factorization is presented [4, 5]. [4] proposed a anti-windup control design method by using H_∞ norm minimization and [5] proposed that by using H_2 norm minimization. [4, 5] are important to give systematic design method of anti-windup contoroller. [4, 5] adopt the method of [11] as the left coprime factorization of the plant $G(s)$. The left coprime factorization $G(s) = \tilde{D}^{-1}(s)\tilde{N}(s), \tilde{D}(s), \tilde{N}(s) \in RH_\infty$ of the plant $G(s)$ has the freedom, for example the direct pass term of $\tilde{D}(s)$ requires to be nonsingular. The left coprime factorization by [11] can not represent the freedom of the left coprime factorization since the direct pass term of $\tilde{D}(s)$ by [11] is fixed to be identity matrix. This implies that [4, 5] does not use freedom of the left coprime factorization.

In the present paper, we consider the effectiveness of using the freedom of the left coprime factorization toward anti-windup phenomenon. At first, we will show both H_2 and H_∞ norm can be arbitrarily small using

the freedom of the left coprime factorization. This implies that H_2 or H_∞ norm minimization of [4, 5] can be solved only using the freedom of the left coprime factorization. Next relation between saturation and settling time of antiwindup control using left coprime factorization method is considered. We show that it is not always that the higher upper bound of control input limitation is, the smaller settling time is.

2   Problem formulation

Let us consider a control system with control input saturation shown in Fig. 1 . Here $G(s)$ is the plant,
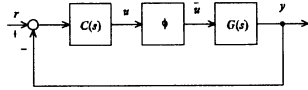
Figure 1: Control system

$C(s)$ is the controller given by

$$C(s) = \left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array}\right] \tag{1}$$

and ϕ is the control input saturation given by

$$\bar{u} = \begin{cases} M & \text{if } u > M \\ u & \text{if } M > u > -M \\ M & \text{if } u < M \end{cases} . \tag{2}$$

$C(s)$ is assumed to satisfy control specification such as

1. to stabilize the plant $G(s)$

2. to hold robust stability condition

3. to satisfy desirable sensitivity characteristics

and so on.

A left coprime factorization of $C(s)$

$$C(s) = (1 - D_c(s))^{-1} N_c(s) \tag{3}$$

is denoted by

$$\begin{cases} D_c(s) = \left[\begin{array}{c|c} A_c - KC_c & K \\ \hline HC_c & 1 - H \end{array}\right] \\ N_c(s) = \left[\begin{array}{c|c} A_c - KC_c & B_c - KD_c \\ \hline HC_c & HD_c \end{array}\right] \end{cases} , \tag{4}$$

where H is nonzero constant real number and K is a constant matrix with appropriate size satisfying $A_c - KC_c$ has no eigenvalues in the closed right half plane. A antiwindup control structure using left coprime factorization is summarized in Fig. 2 . Several papers give antiwindup control design method in Fig.
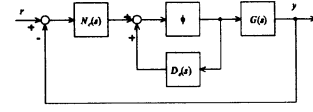
Figure 2: Antiwindup control using left coprime factorization method

2 . [4] proposed a anti-windup control design method by using H_∞ norm minimization of $\|N_c(s)\|_\infty$ and [5] proposed that by using H_2 norm minimization of $\|N_c(s)\|_2$. But [4, 5] does not use the freedom of H that is the freedom of left coprime factorization. From (4), it is obvious that both $\|N_c(s)\|_\infty$ and $\|N_c(s)\|_2$ can be arbitrary minimized using H. This implies that if $\|N_c(s)\|_\infty$ and $\|N_c(s)\|_2$ are suitable for showing the effect of antiwindup, the antiwindup characteristics can be obtained using only H.

In the present paper, we consider

1. the effectiveness of using the freedom of the left coprime factorization toward anti-windup phenomenon

2. relation between saturation and settling time of antiwindup control using left coprime factorization method

3   Effect of H

In this section, the effectiveness of using the freedom of the left coprime factorization toward antiwindup phenomenon is considered using a numerical example. Effects of the antiwindup appear the overshoot and the settling time. Therefore we measure an effect of the antiwindup by the overshoot and the settling time.

We consider a plant

$$G(s) = \frac{500}{s^3 + 5s^2 + 107s + 303} \tag{5}$$

to measure the effectiveness of using the freedom of the left coprime factorization toward anti-windup phenomenon. When the plant is asymptotically stable, the control structure of the left coprime factorization is similar to the Internal Model Control. The structure of is shown in Fig. 3 .

When $Q(s)$ is denoted by

$$Q(s) = G_m^{-1}(s)q(s), \tag{6}$$

where $G_m(s)$ and $q(s)$ are deffined

$$G_m(s) = HG(s) \tag{7}$$
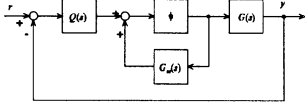
Figure 3: Internal Model Control

$$q(s) = \frac{70^3}{(1 + 70s)^3}, \qquad (8)$$

respectively.

When upper bound and lower bound of control input saturation M in (2)is settled by $M = 5$, the step response of the control system in Fig. 3 and the control input is shown in Fig. 4 and Fig. 5 , respectively. When H is changed between 0.01 and 1,
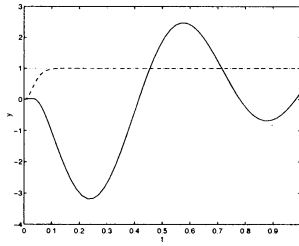
Figure 4:
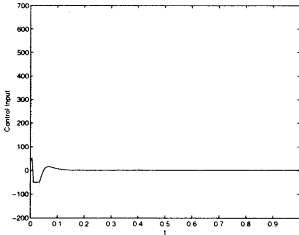
Figure 5:

the peak value of overshoot, that is maximum value of the output is shown in Fig. 6 . When H is changed between 0.01 and 1, the settling time is shown in Fig. 7 . When $M = 2$, the settling time is shown in Fig. 8 . From above figures, we can conclude

1. the freedom of the left coprime factorization has lower effectiveness toward to the antiwindup

2. from Fig. 7 and Fig. 8 , we can not persist that the larger M is, the smaller settling time is.
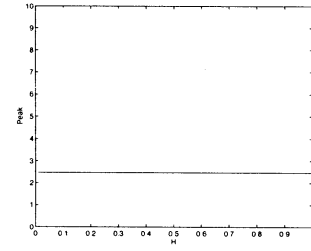
Figure 6:
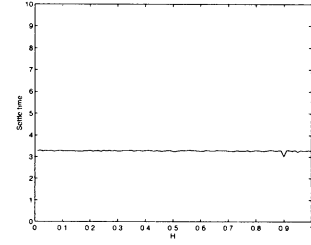
Figure 7:

4 Relation between saturation and settling time

In this section, we will consider relation between saturation and settling time of antiwindup control using left coprime factorization method. From Fig. 7 and Fig. 8 , we can not persist that the larger M is, the smaller settling time is. Next the relation between saturation and settling time of antiwindup control using left coprime factorization method using same numerical example in previous section.

The relation between upper bound M of control input saturation and the settling time is shown in Fig. 9 and Fig. 10 . This figure shows following expressions:
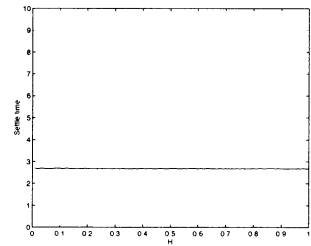
Figure 8:
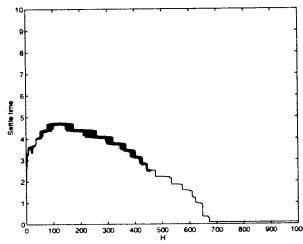
Figure 9:
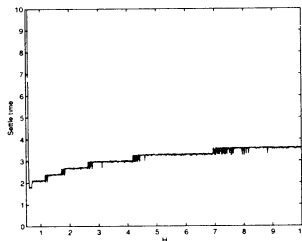
Figure 10:

1. we can not persist that the larger M is, the smaller settling time is.

2. if the upper bound of control input saturation is greater the certain value, as the upper bound of control input saturation as increase, the settling time decreases monotonically, that is, if do not, the settling time does not always decrease monotonically.

3. The settling time changes stepwise toward the upper bound of control input saturation M.

5 Conclusion

In the present paper, we considered the effectiveness of using the freedom of the left coprime factorization toward anti-windup phenomenon. At first, we showed both H_2 and H_∞ norm can be arbitrarily small using the freedom of the left coprime factorization. The freedom of left coprime factorization can not use the antiwindup Next relation between saturation and settling time of antiwindup control using left coprime factorization method was considered. We showed that it is not always that the higher upper bound of control input limitation is, the smaller settling time is.

References

[1] H.J.Sussman, E.D.Sontag and Y.Yang : A General Result on the stabilization of linear systems using bounded controls, IEEE Trans. On AC, AC-39-12, 2411/2425(1994)

[2] W.Liu, Y. Chitour and E.Zontag : On finite - gain stabilizability of linear systems subject to input saturation, SIAM J. on Control and Optimization, Vol.34-4,1190/1219(1996) Robust Process Control,(1989) Prentice Hall

[3] M. Morari : Some control problems in the Process Industries, In H.L.Trentelman and J.c.Willems(Eds), Essays on Control: Perspectives in the Theory and its Applications, Birkhser, 55/78, (1993)

[4] R. Watanabe, A. Matsui, K.Uchida and E.Shimemura : Design of Anti-windup controller based on left coprime factorization, Proc. 16th Symposium on Dynamical System Theory, 57/62(1993)(in Japanese)

[5] M. Saeki, M. Tsuchiya and N. Wada : Design of anti-windup controller by H_2 norm minimization, Proc. 26th symposium on control theory, 183/188(1997)(in Japanese)

[6] R. Hanus, M. Kinnaert and J.L.Henrotte : Conditioning Technique, a General Anti-windup and Bumpless Transfer Method, Automatica, 23-6, 729/739(1987)

[7] S.Miyamoto, G.Vinnicombe : Robust control of plant with saturation nonlinearity based on coprime factor representation, Proc. CDC35,2838/2840(1996)

[8] J.C.Doyle, R.S.Smith and D.F.Enns : Control of plants with input saturation nonlinearities, American Control Conference, 1034/1309(1987)

[9] K. Yamada : Robust Internal Model Servo Control with Control Input Saturation, Proc. 1998 American Control Conference, 3685/3686(1998)

[10] Y. Funami and K. Yamada: An anti-windup control design method using Modified Internal Model Control Structure, IEEE conf. SMC'99 to be published

[11] C.N.Nett, C.A.Jacobson and M.J. Balas: A connection between satate-space and doubly coprime fractional representation, IEEE Trans. Automatic Control, AC29-9, 831/832(1984)

Modeling of Human Motor Control on Manual Tracking to Moving Visual Target Based on Neurophysiological Background

T. Sugi*, M. Nakamura**, J. Ide† and H. Shibasaki ‡

*Department of Electrical and Electronic Engineering, Saga University, Saga 840-8502, Japan

** Department of Advanced Systems Control Engineering, Saga University, Saga 840-8502, Japan

† Department of Commerce, Seinan Gakuin University, Fukuoka, 814-8511, Japan

‡ Department of Brain Pathophysiology, Kyoto University, Kyoto 606-8501, Japan

E-mail : sugi@cntl.ee.saga-u.ac.jp

Abstract

Several regions in human brain participate in executing the delicate and smooth motor control. Various movement disorders are caused according to the damaged regions in the human brain related to the motor control. In this paper, mathematical model of the human motor control on manual tracking to moving visual target was constructed based on neurophysiological background on movement disorders. Parameters in the proposed model were determined with respect to the actual data of the manual tracking test for the normal healthy adults and patients with movement disorders. Relationship between the model parameters and the characteristics of the movement disorders was analyzed for verifying the proposed model.

Keywords: Human Motor Control, Visual Tracking Test, Mathematical Model, Neurophysiological Background

1 Introduction

The control of a voluntary movement of human hand is significantly associated with the cerebral cortex, the cerebellum and the basal ganglia in parts of the human brain. Each portion related to the motor control has an individual role for the accurate movement of the hand. Therefore, the disorders for respective regions bring about the influences for control of hand motion. Visual tracking test is one of the effective method for analyzing the relationship between the hand movement and movement disorders [1]. On the other hand, studies on modelling of human arm motion control [2], controller design based on the human brain function [3], and so on have been done.

In this study, a mathematical model of the human motor control for manual tracking of visual target was constructed in accordance with the neurophysiological background on movement disorders. Statistical properties of experimental data for normal healthy adults and patients with movement disorders were used for determination of the model parameters. Characteristics of the model obtained from the results were successfully conformed with that of patients with movement disorders on neurophysiological knowledges.

2 Manual Tracking to Moving Visual Target

The measurement equipment of visual tracking test used in this study consists of personal computer, digitizer, pen and CRT monitor [1]. A subject holds a pen on his/her hand and moves it on the digitizer. Position of the hand with a pen was displayed on the monitor as a green rectangular point. The visual tracking tests were carried out for 11 subjects (3 normal adults, 4 patients with Parkinson disease and 4 patients with cerebellar ataxia). The task given for the subjects was to pursue a red circled visual target which moved on a straight line vertically with a constant speed (3 cm/s) on a CRT monitor. Visual information of the objective point and the hand movement (pursuit point) was acquisited from the display of the measurement equipment.

3 Mathematical Model of Human Motor Control for Manual Tracking to Visual Target

3.1 Model structure

A mathematical model of the human motor control on manually tracking to the visual target was

constructed based on neurophysiological knowledges. Fig. 1(a) shows the signal flow on manual tracking to the visual target. First, positions of a hand and a target on the monitor are aquisited from the retina (eyes) and the visual cortex. Next, cerebral cortex, cerebellum and basal ganglia calculate the control input of a hand based on the visual information. Finally, control input is transfered to the musculoskeletal system of a hand though the motor area (MI) and lower motor neuron, and the hand tip motion is reflected on the monitor again. We assumed that the motor control for manual tracking to visual target was done by using only the visual information, and other information such as somatosensory was negligible.

Fig. 1(b) shows the proposed mathematical model of visual motor control displayed in Fig. 1(a). The proposed model consists of three parts ; motion ideation part, motion planning part and motion execution part. The motion ideation part, corresponding to the association cortex and visual cortex in the brain, roles a decision making of the movement according to the visual information. The motion planning part which corresponds to the cerebral cortex, cerebellum and basal ganglia, calculates the control inputs for hands based on visual information. In the proposed model, motion planning part was divided into two blocks according to their roles in motor control. The motion execution part which corresponds to the motor area in the cerebral cortex, lower motor neuron and musculoskeletal system (hand), executes the actual motion according to the control inputs generated in the motion planning part. In this study, dynamical properties of a human hand was expressed by using that of 2-link robot arm.

3.2 Motion ideation part

Role of retina and visual cortex is to obtain the informations which are necessary to generate the control input for a hand. All visual informations are calculated from the hand tip position $X_h = [x_h, y_h]^T$ and the position of the visual target $X_r = [x_r, y_r]^T$ displaying on the monitor. Necessity visual informations for motion planning were assumed as the position and velocity of a hand and a target. Visual information vector $V_I = [\ x_h, y_h, x_r, y_r,\ \dot{x}_h, \dot{y}_h, \dot{x}_r, \dot{y}_r\ ]^T$ was expressed as

$$V_I = H_h X_h + H_r X_r \qquad (1)$$

$$H_h = \begin{bmatrix} 1 & 0 & 0 & 0 & \frac{d}{dt} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \frac{d}{dt} & 0 & 0 \end{bmatrix}^T$$

$$H_r = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & \frac{d}{dt} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \frac{d}{dt} \end{bmatrix}^T$$

In the association cortex, volition for executing the motion occures according to the given task, and the ideation for motion planning genarates by combining the volition and the visual information. The role of association cortex was not taken into account in the proposed model.

3.3 Motion planning part

Informations getting from the monitor of the measurement equipment are the visual coordinates expressed by the XY-coodinate. However, the actual information for genarating the control input of a hand motion can not be considered as visual coordinate but physical coordinate of a human body. Accordingly, the visual information vector V_I was changed from the XY-coodinate to the joint coordinate. Position and velocity of the hand tip were converted to the joint angle of the shoulder θ_{1h} and the elbow θ_{2h}, and their joint angular velocity $\dot{\theta}_{1h}$ and $\dot{\theta}_{2h}$. Modified visual information vector $V_J = [\theta_{1h}, \theta_{2h}, \theta_{1r}, \theta_{2r}, \dot{\theta}_{1h}, \dot{\theta}_{2h}, \dot{\theta}_{1r}, \dot{\theta}_{2r}]^T$ was calculated by inverse kinematics and the inverse *Jaccobian* matrix.

Cerebral cortex and cerebellum are the main regions for generating the actual control input for a hand. If the cerebellum has damaged such as patient with cerebellar ataxia, jerky and non-smooth motion appeare as movement disorders. In the proposed model, role of the cerebral cortex and the cerebellum was regarded as making the control input for each joint of a hand from the visual information V_J. The torque input $\tilde{\tau}$ $(=[\tilde{\tau}_1, \tilde{\tau}_2]^T)$ for each joint was calculated as

$$\tilde{\tau} = e^{-L_M s} e^{-L_C s} K V_J \qquad (2)$$

$$K =$$

$$\begin{bmatrix} -k_p & 0 & k_p & 0 & -k_v & 0 & k_v & 0 \\ 0 & -k_p & 0 & k_p & 0 & -k_v & 0 & k_v \end{bmatrix} \qquad (3)$$

where L_M means the time delay for calculating the control input of a hand, L_C expresses the time delay originating from the movement disorder in the cerebellum. k_p and k_v show the feedback gain for position error and velocity error.

Basal ganglia participates in the sequential motor control by cooperating with the supplementary motor area and the motor area. Parkinson disease is caused by the damage in the basal ganglia and its characteristics of the movement disorder is the delay of timing and reduction in the motion. Acoording to the characteristics of the movement disorder on the basal ganglia, the final control input $\tau (= [\tau_1, \tau_2]^T)$ was calculated as

$$\tau = e^{-L_B s} B \tilde{\tau} \qquad (4)$$
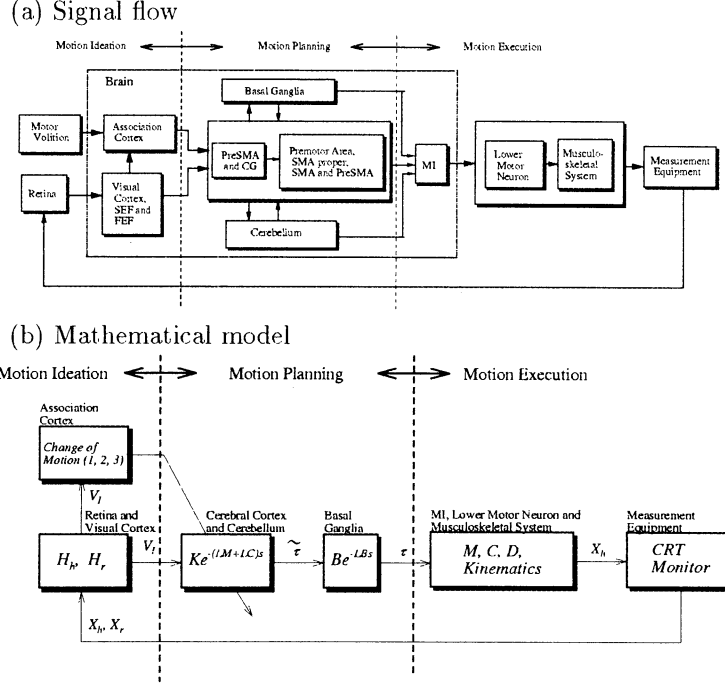
(a) Signal flow

Figure 1: (a) Signal flow on manual tracking to visual moving target. (b) Mathematicalmodel on manual tracking to visual moving target. Proposed model consists of three blocks ; motion ideation part, motion planning part and motion execution part.

$$B = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix} \qquad (5)$$

where, L_B is a time delay caused by the disorder for the basal ganglia and component b in B matrix expresses the reduction of the control input. Parameter b ranged from 0 to 1.

3.4 Motion execution part

Dynamics of a human hand in the visual tracking test was expressed by 2-link SCARA type robot arm. Equation for expressing the dynamical properties of robot arm can be written as

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + D(\dot{\theta}) = \tau \qquad (6)$$

where, $\tau(= [\tau_1, \tau_2]^T)$ the joint torque, $\theta(= [\theta_1, \theta_2]^T)$ the joint angle, M the inertia, C the centrifugal force and D the friction, respectively. In actual, the hand tip position reflects on the display of the measurement equipment so that the hand tip position was obtained from the joint angle of a hand by using the kenematics as

$$x = l_1 \cos\theta_{1h} + l_2 \cos(\theta_{1h} + \theta_{2h}) \qquad (7)$$
$$y = l_1 \sin\theta_{1h} + l_2 \sin(\theta_{1h} + \theta_{2h}) \qquad (8)$$

where l_1 the length of the upper arm and l_2 the length of the forearm.

3.5 Determination of the model parameters by use of measurement data

The parameters in the model were appropriately determined by use of the experimental data of the visual tracking test for three groups. Characteristics of the hand position, velocity and acceleration were reflected to the model parameters in the motion planning part.

4 Result and Discussion

Table 1 shows the parameters determined by the actual data of the normal subject, cerebellar ataxia and parkinson disease.

Validity of the constructed model was inspected through the simulation study. Fig. 2 shows the comparison of the hand movement between the actual data and the model output. Broken line expresses the visual target, gray solid line indicates the hand tip motion of the actual data and black solid line means the model output. Fig. 2 shows the typical examples for normal
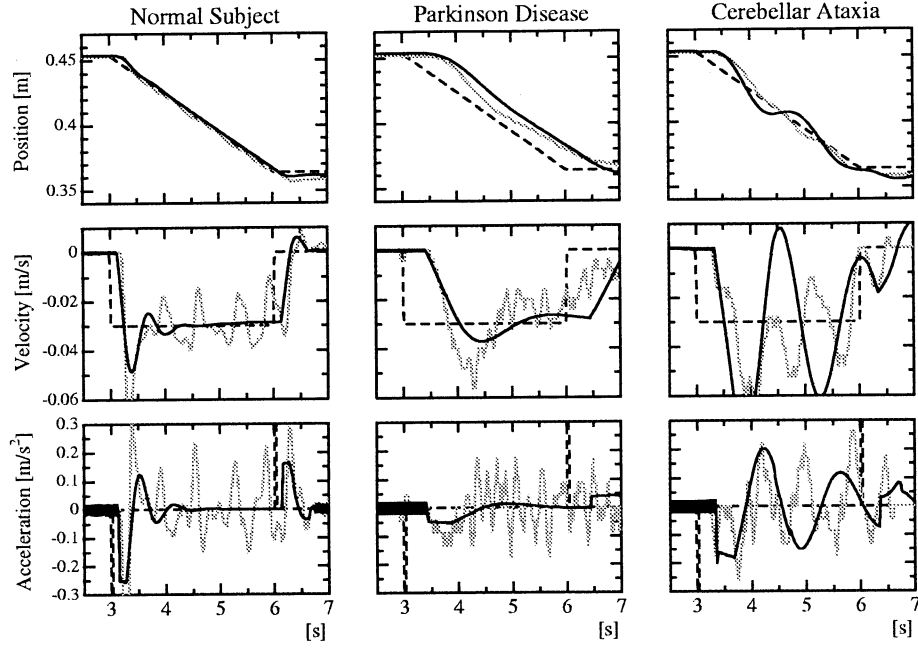
Figure 2: Comparison of charcteristics of manual tracking to visual moving target between the proposed model and actual data.

Table 1: Parameters corresponding to the role of respective regions in human brain. All parameters were determined with respect to the actual data of the manual tracking test for the patients with movement disorders.

Group	Time delay			K		B
	L_M	L_C	L_B	k_p	k_v	b
Normal Subject	0.1	0.0	0.0	0.05	0.15	1.00
Cerebellar Ataxia	0.1	0.2	0.0	0.05	0.10	1.00
Parkinson disease	0.1	0.0	0.3	0.05	0.15	0.25

subject (left side), patient with cerebellar ataxia (middle) and patient parkinson disease (right side). Quick reaction and smooth tracking was seen in the data of normal subject. Slow movement was seen in the data of parkinson disease. Jerky and non-smooth motion was seen in the data of cerebellar ataxia. The behavior of the constructed model was in a good agreement with the properties of respective groups.

Relationship between the value of parameters and the control performance of hand movement showed the specific role of respective portions in the brain, especially for the cerebellum and basal ganglia. This results represented that the cerebellum and the basal ganglia played an important role in the motor control for manual tracking to visual target.

5 Conclusion

Results obtained through the proposed model was in a good agreement to something by chance the neurophysiological knowledges on characteristics of the motor behavior of patients. The findings through this study gives an effective information for controller design of robot arm whose movement is similar to human hand movement.

References

[1] Nakamura M, Sugi T, Shibasaki H, Terada K (1993), A method for analyzing characteristics of manual tracking of visual target by using artificial neural network (in Japanese). Jpn. J. MEBE, 31-3, pp.245-253

[2] Sanner RM, Kosha M (1999), A mathematical model of the adaptive control of human arm motions. Biological Cybernetics, 80, pp.369-382

[3] Kawato M, Furukawa K, Suzuki R (1987), A hierarchical neural-network model for control and learning of voluntary movement. Biological Cybernetics, 57, pp.169-185

Path Planning and Navigation of a Mobile Robot as Discrete Optimization Problems

Harukazu IGARASHI and Kiyoshi IOI

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

In this paper, we propose a new solution to path planning and navigation of a mobile robot. In our approach, we formulated the following two problems at each time step as discrete optimization problems: 1) estimation of a robot's location, and 2) action decision. For the first problem, we minimize an objective function that includes a data term, a constraint term and a prediction term. This approach is an approximation of Markov localization. For the second problem, we define and minimize another objective function that includes a goal term, a smoothness term and a collision term. Simulation results show the effectiveness of our approach.

Key Words: Path planning, Navigation, Mobile robot, Optimization problem, Markov localization

1 Introduction

There are many works on the navigation [1] and path planning [2] of mobile robots. These efforts have been made to relieve human operators from the need to constantly specify a robot's motion. However, there is a huge diversity among the navigation and path planning problems in the real world because of the enormous number and great variety of assumptions on the environments, constraints and tasks imposed on a robot.

For navigation, estimating the present location of a robot is the most important problem. However, there is no universal solution to this problem due to the diversity mentioned above. What kind of sensor data can a robot use? If it can use several sensors, how does it fuse these data? Moreover, which direction can a robot move toward? If the robot can only move in the forward direction, we have to impose a constraint that the robot direction should be the same direction of its previous movement. Of course, a dead reckoning approach is a very effective one. However, a robot must have a map of its environment, and the error in the robot's movement should be minimal if a dead reckoning approach is taken.

For path planning, the problem can be either static or dynamic, depending on whether the information on the robot's environment is fixed or capable of being updated. It also depends on whether there are constraints on the robot's motion. This includes bounds on the robot's velocity and acceleration and constraints on the curvature of the robot's paths. Moreover, different tasks require different criteria for estimating paths or trajectories. For example, users of an automatic motion planner may require the shortest path or the safest path

depending on the task the user wants to make the robot do.

This paper proposes a solution to deal with this diversity in path planning and navigation. We formulate the following two problems at each time step as two discrete optimization problems: 1) estimation of a robot's location, and 2) action decision. In our approach to the first problem, an objective function to minimize includes a data term, a constraint term and a prediction term. This approach is shown to be an approximation of Markov localization. For the second problem, we define another objective function to minimize that includes a goal term, a smoothness term and a collision term. Simulation results show the effectiveness of our approach.

2 Navigation

In our approach, the navigation algorithm is expressed as a loop consisting of the following five steps: i) sensing the robot's environment, for example, distances to obstacles or walls; ii) estimating the robot's location; iii) planning and deciding the robot's next action; iv) carrying out the action; and v) checking whether the stop condition has been satisfied. This basic loop for navigation is shown in Fig. 1. The second and third steps are particularly important research subjects. We consider these two processes as different discrete optimization problems.

We will first define our navigation problem and robot in detail before defining the objective functions and solution spaces of the two optimization problems. Let us assume the following points. The objective of navigation is to navigate a robot among static obstacles. The robot has no body but has several sensors distributed around it radially.
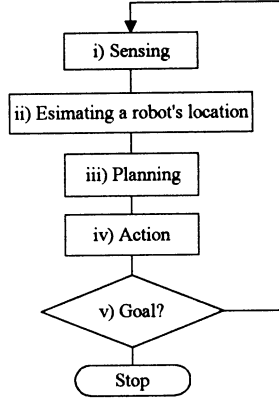
Fig.1 The basic algorithm of navigation

Moreover, the robot can use a map that describes the precise positions and shapes of obstacles.

3 Localization of a robot

We define the following objective function E_1 of a robot's position r_t and 2D direction α_t at a discrete time t. The minimal solution of Eq. (1) gives the estimates for the robot's location at time t.

$$E_1\left(r_t, \alpha_t; R_t^{obs}, r_{t-1}, v_{t-1}\right) \equiv a_1 E_{data} + a_2 E_{cnst} + a_3 E_{prdt} \quad (1)$$

The three terms in the right-hand side of Eq. (1) are a *data term*, a *constraint term*, and a *prediction term*, respectively.

The data term E_{data} is the disparity between the sensed range data R_t^{obs} at time t and the data pattern $R_{map}(r_t, \alpha_t)$ that should be observed if the robot stands at point r_t and faces direction α_t. This disparity is measured by the following square distance between the two vectors:

$$E_{data}\left(r_t, \alpha_t; R_t^{obs}\right) \equiv \left\| R_{map}(r_t, \alpha_t) - R_t^{obs} \right\|^2 \quad , \quad (2)$$

where $\|X\|$ means a Euclid norm of a vector X.

For a robot that can only move in the forward direction, the constraint term E_{cnst} represents the constraint that direction $Dir(\Delta r_t)$ of vector $\Delta r_t = r_t - r_{t-1}$, which is the robot's movement between times t-1 and t, should agree with the robot's forward direction α_t. We represented this constraint term by the following disparity.

$$E_{cnst}\left(r_t, \alpha_t; r_{t-1}\right) \equiv \begin{cases} \left[\alpha_t - Dir(r_t - r_{t-1})\right]^2 & \text{if } r_t \neq r_{t-1} \\ 0 & \text{if } r_t = r_{t-1} \end{cases} \quad (3)$$

The prediction term E_{prdt} is a disparity between location (r_t, α_t) and location (r'_t, α'_t). The location (r'_t, α'_t) is a vector predicted for (r_t, α_t) from the preceding location (r_{t-1}, α_{t-1}) and the preceding action v_{t-1}. This disparity is defined by

$$E_{prdt}\left(r_t, \alpha_t; r'_t, \alpha'_t\right) \equiv \left(r_t - r'_t\right)^2 + \left(\alpha_t - \alpha'_t\right)^2, \quad (4)$$

where $(r'_t, \alpha'_t) = (v_{t-1}\Delta t + r_{t-1}, Dir(v_{t-1}))$. Eq. (4) shows that the data (r_t, α_t) estimated for the robot's location

at time t by minimizing E_1 depends on the action, v_{t-1}, at the preceding time. This estimated data (r_t, α_t) is used for deciding the next action of a robot.

4 Relation with Markov localization

In this section, we show that our approach using Eq. (1) is an approximation of a Markov localization algorithm [3].Following the same symbols used by the research group, Burgard et al., let $l = (r, \alpha)$ denote a robot's location. Next, let $Bel(l)$ be the robot's subjective belief that it is at location l. The function $Bel(l)$ takes a value in [0,1] and satisfies the normalization condition that $\int Bel(l)dl = 1$.

In the Markov localization algorithm, $Bel(l)$ is updated whenever (i) the robot moves as

$$Bel(l) = \int P_a(l|l')Bel(l')dl', \quad (5)$$

where $P_a(l|l')$ denotes the probability that action a carries the robot to l when executed at l', and (ii) the robot senses as

$$Bel(l) \leftarrow \frac{P(s|l)Bel(l)}{P(s)}, \quad (6)$$

where s is a sensor reading and $P(s|l)$ is the likelihood of perceiving s at l. $P(s|l)$ can be calculated from a map of the robot's environment.

Let $Bel_a(l)$ denote a belief function after taking an action a. If the robot carries out action a, then the update rule of Eq. (5) leads to the relation that

$$Bel_a(l) = \int P_a(l|l')Bel(l')dl'. \quad (7)$$

After sensor reading, $Bel_a(l)$ is updated to $Bel_{a,s}(l)$ as

$$Bel_{a,s}(l) = \frac{P(s|l)}{P(s)} \int P_a(l|l')Bel(l')dl' \quad (8)$$

by substituting $Bel_a(l)$ of Eq. (7) into $Bel(l)$ in the right-hand side of Eq. (6). If we use only the estimated result l_0 for the robot's location l at the preceding cycle in $Bel(l')$ of Eq. (8), we obtain that

$$Bel_{a,s}(l) \approx \frac{P(s|l)}{P(s)} P_a(l|l_0) \quad (9)$$

$$= \frac{P(s|l)}{P(s)} P_a(r|l_0) P_a(\alpha|r, l_0). \quad (10)$$

Comparing Eq. (10) with Eq. (1), we can find a correspondence between maximizing $Bel_{a,s}(l)$ and minimizing E_1, where the three conditional probabilities $P(s|l)$, $P_a(r|l_0)$ and $P_a(\alpha|r, l_0)$ correspond to E_{data}, E_{prdt} and E_{cnst}, respectively. In MAP estimations (Maximum a posteriori estimation), a problem of maximizing a product of conditional probabilities is often formulated as a problem of minimizing an objective function [4].

Consequently, our approach of minimizing E_1

in Eq. (1) to estimate a robot's location is an approximation of Markov localization. It reduces computation time because it does not carry out the integral operation in the right-hand side of Eq. (8).

5 Path planning

This section describes our approach to path planning. We assumed that a series of actions decided at each time point would derive a desirable path for a robot. We define the following objective function E_2 of a velocity v_t at time t,

$$E_2\left(v_t;r_t,v_{t-1},r_{goal}\right) = b_1 E_{goal} + b_2 E_{smth} + b_3 E_{clsn}. \quad (11)$$

The minimal solution of Eq. (11) gives an estimate for the robot's action at time t.

The first term in the right-hand side of Eq. (11), E_{goal}, represents an attractive force to the goal r_{goal}. It is defined as

$$E_{goal}(v_t;r_t,r_{goal}) \equiv \text{sgn}[G(v_t)] \cdot G(v_t)^2, \quad (12)$$

where sgn(x) denotes the sign of x and

$$G(v_t) \equiv \left\| r_{goal} - r'_{t+1}(v_t;r_t) \right\| - \left\| r_{goal} - r_t \right\|. \quad (13)$$

The second term in Eq. (11), E_{smth}, is defined as

$$E_{smth}\left(v_t;v_{t-1}\right) = \left\| v_t - v_{t-1} \right\|^2 \quad (14)$$

to minimize a change in a robot's velocity vector.

The last term in Eq. (11), E_{clsn}, represents a repulsion force for avoiding collisions with obstacles and walls. We define the term as

$$E_{clsn}\left(v_t;r_t\right) = \begin{cases} D_{clsn} & \text{if } Dist\left(r'_{t+1}\right) < 0 \\ -Dist\left(r'_{t+1}\right)^2 & \text{if } 0 < Dist\left(r'_{t+1}\right) < R \\ -R^2 & \text{if } Dist\left(r'_{t+1}\right) > R \end{cases}, \quad (15)$$

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

Thrun et al. proposed another optimization approach for motion planning [5]. In their approach, a non-Markov term, such as E_{smth} shown in Eq. (14), was not taken into account.

6 Simulation

6.1 Locations

We applied the approach proposed in Sections 2 and 3 to a static problem, where the information on the robot's environment is fixed and known.

Figure 2 shows locations of obstacles, walls, the
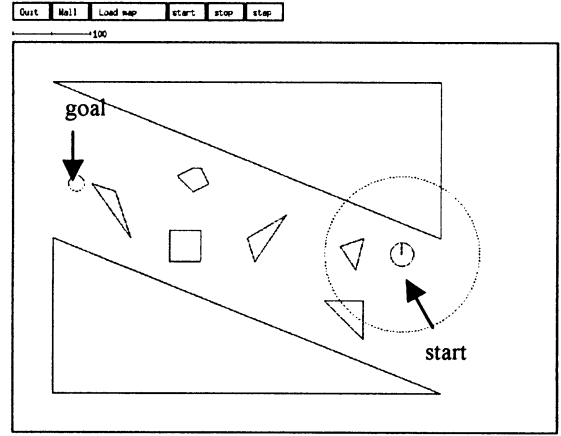
Fig.2 Obstacles, walls and the initial location of the robot

start point and the goal point of a robot. In Fig. 2, a short circle and a short line drawn from the circle's center mean the robot's position and front direction, respectively. Please note that the robot has no body and that the short circle and line in Fig. 2 were used as a matter of convenience to show the robot's location. However, the robot has sixteen range sensors placed around it radically. The range of sensors was set at 100 and is shown as a circle on a dotted line in Fig. 2.

6.2 Search spaces

We restricted and discretized search spaces when minimizing the functions E_1 and E_2. In minimizing E_1, we searched areas around the location (r'_t, α'_t) predicted from the previous location and action. In minimizing E_2, we only took into account velocity vectors whose lengths are smaller than a constant. Figures 3 and 4 show search spaces used in our simulation for minimizing E_1 and E_2, respectively.

6.3 Results

In our approach, results depend on weights $\{a_i\}$ and $\{b_i\}$ (i=1,2,3). We show several simulation results for four cases: (i) $a_1=1$, $a_2=a_3=0.1$, $b_1=1$, $b_2=0.1$, $b_3=10$; (ii) $a_1=a_2=0,a_3=0.1$, $b_1=1$, $b_2=0.1$, $b_3=10$; (iii) $a_1=1,a_2=a_3=0.1$, $b_1=10$, $b_2=0.1,b_3=10$; and (iv) $a_1=1,a_2=a_3=0.1,b_1=1,b_2=0.5,b_3=10$. The trajectories obtained in the four cases are shown in Figs. 5 to. 8. In case (i), the robot reached the goal at t=31. There
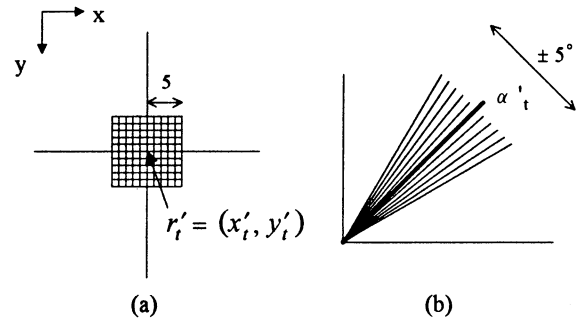
Fig.3 The area searched for a robot's (a)position and (b)posture
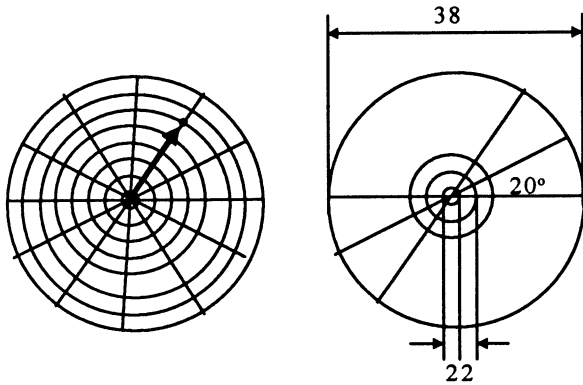
Fig.4 The area searched for the optimal velocity of a robot

is little difference in the estimated locations and the actual locations of a robot (Fig. 5). The difference was observed in case (ii), where only the prediction term E_{prdt} was considered in E_1. That is estimation by dead reckoning with a map. However, the fact that there was no perceptional error in case (i) shows that the estimation errors in Fig. 6 were corrected by the data term and the constraint term.

In case (iii), the goal term is increased by ten times that of case (i). As a result, the robot reached the goal at t=28, which is earlier than in case (i), while it also often ran into obstacles (Fig. 7).

If the smoothness term is increased by five times in case (i), a straight trajectory as shown in Fig. 8 was obtained. In cases (iii) and (iv), the estimation error of a robot's location was almost unobserved because the error correction by E_{data} and E_{cnst} worked well, as it did in case (i).

7 Discussion

Our simulation showed that trajectories depend on weights in E_2. Therefore, weights have to be controlled properly to get a desirable trajectory. Accordingly, we plan to apply a reinforcement learning similar to REINFORCE [6] to our action decision.

References

[1] Kortenkamp, D., Bonasso, R.P., and Murphy, R. (eds.), Artificial Intelligence and Mobile Robots. AAAI Press/The MIT Press (1998).
[2] Hwang,Y.K. and Ahuja,N.(1992),Gross Motion Planning—A Survey. ACM Computing Surveys 24(3): 219-291
[3] Burgard, W., Fox, D. and Thrun, S. (1997), Active Mobile Robot Localization. In: Proceedings of 15th Joint Conference on Artificial Intelligence (IJCAI97), Nagoya, Japan, August 23-29, 1997, Vol.2, pp. 1346- 1352
[4] Geman, S. and Geman, D. (1984), Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. & Mach. Intell., PAMI-6, pp. 721-741
[5] Thrun, S. et al., Map Learning and High-Speed Navigation in RHINO, in [1], pp. 21-52.
[6] Williams, R.J. (1992), Simple Statistical Gradient-

Following Algorithms for Connectionist Reinforcement Learning. Machine Learning 8: 229-256
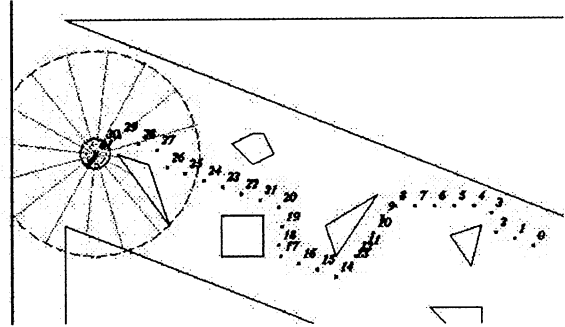
Fig.5 The robot's trajectory in case(i) when $a_1=1$, $a_2=a_3=0.1$, $b_1=1$, $b_2=0.1$ and $b_3=10$. The numbers shown here are cycle numbers
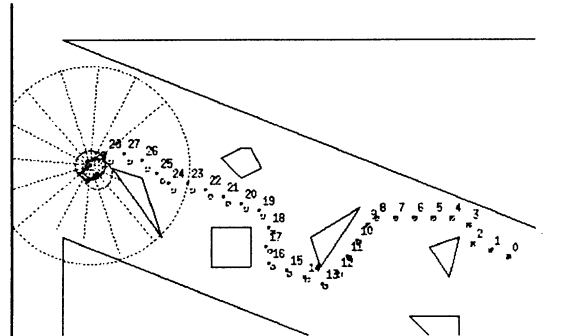
Fig.6 The robot's trajectory in case(ii) when $a_1=a_2=0$, $a_3=0.1$, $b_1=1,b_2=0.1$ and $b_3=10$. The series of symbol □ denote actual positions
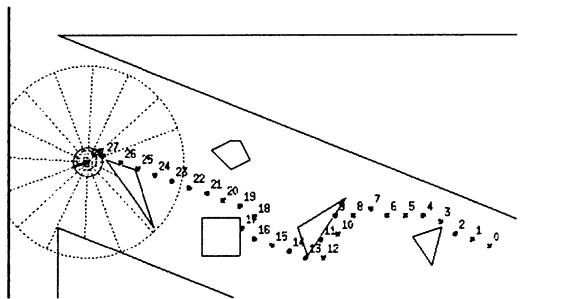
Fig.7 The robot's trajectory in case(iii) when $a_1=1$, $a_2=a_3=0.1$, $b_1=10$, $b_2=0.1$ and $b_3=10$
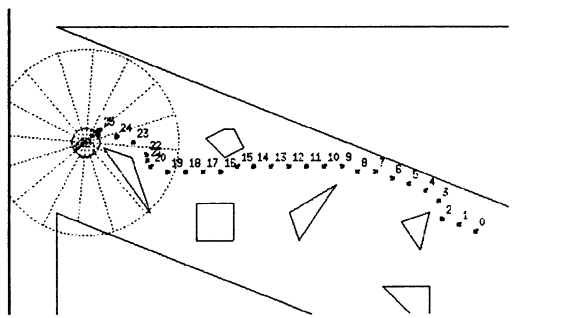
Fig.8 The robot's trajectory in case(iv) when $a_1=1$, $a_2=a_3=0.1$, $b_1=1$, $b_2=0.5$ and $b_3=10$

High-Level Spatial Scenarios in WAVE

Peter S. Sapaty

Institute of Mathematical Machines and Systems
National Academy of Sciences of Ukraine, Glushkova Avenue 42, 252187 Kiev, Ukraine
Voice: +380-44-2665023, Fax: +380-44-2666457, sapaty@immsp.kiev.ua

Abstract. The paper describes an extended WAVE model for an efficient parallel and distributed processing in a variety of dynamic physical and virtual environments, as well as in their unity. It allows us to receive integral system solutions in the form of high-level, self-evolving spatial scenarios, which are much superior to the traditional agent-based organizations. The model may underlie advanced networking and multiple robotic systems oriented on solving complex distributed exploration, optimization, control, simulation, and decision making problems cooperatively.

Keywords: parallel and distributed computing, spatial pattern-matching, mobile agents, mobile robots, networking, WAVE language

1 Introduction

Computers have moved into space. They received bodies. What they have to do now is much greater and more diverse than mere computations. They manipulate complex objects, deliver goods, patrol warehouses, demine territories, dismantle faulty reactors, fight forest fires, saving human health and lives.[1] Stationary or mobile, embodied or bodyless, computers are inter-linked, networked. They may need to operate collectively in solving complex problems, requiring efficient coordination, and in special cases strong overall control. The computer networks may have dynamic and open nature, changing composition and communication structure at run time. In some sense, the whole world may be considered as one networked computer processing dynamic distributed information in a highly parallel mode.

How to manage these extended multifunctional applications of computers and cope with an explosive diversity of programming models, languages and techniques, inevitably leading to numerous seams and patches in the complex system design, and huge overhead? The only way is *to change radically the basic computational model*, making it primarily oriented on parallel, distributed and cooperative system organization, management and control, rather than computations. It should also occupy an essentially higher system conceptual level, in order to be able to hide within implementation a multitude of traditional organizational routines.

WAVE [2,3] represents such class of a model which is very much close to the holistic, or gestalt, psychology and theory. In traditional approaches, the overall system behavior is considered to be a derivative of work of many parts, or "agents" and interactions between them,[4] and most of distributed programming languages and techniques are just oriented on a detailed description of these parts and exchange of messages between them, as well as movement of agents themselves.[5]

WAVE, on the contrary, offers a unique possibility of programming the desired whole behavior as a starting point, in the form of high-level active spatial scenario or pattern. Particular hardware or software agents and their interactions emerge dynamically, on the implementation level, if and when required or available, during the pattern's wavelike stepwise, intersectional coverage of space. This being supported by recursive track-based echo control (which can be lifted), dynamic stationary and mobile variables, distributed spatial backtracking and looping (see Fig. 1).
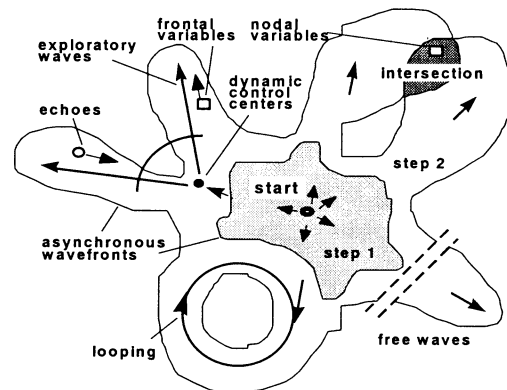
Fig. 1 Controlled flooding, coverage of space in WAVE

Operation of WAVE in a variety of virtual spaces has been researched,[2] and its use for cooperative physical world scenarios investigated.[3] The current paper generalizes the model to operate efficiently in a unity of physical, virtual, and execution worlds (respectively, PW, VW, and EW), providing its universality and extended applications.

2 Representation of Space in WAVE

Physical world. Any point, or *node*, in a continuous PW may be represented and reached by its absolute coordinates. Moving to other nodes can use the destination coordinates again, or coordinate shifts from the previous node, speed value may be added, if appropriate. As a PW space is

continuous, and coordinates have limited precision, any new node reached is considered unique. It is, however, possible to move to the already existing nodes, reached by other activities. For this, the destination coordinates are defining a search start position, search radius is also added within which the existing nodes may reside. More than one node (including all) may be reentered simultaneously in such a way, depending on the radius value. PW nodes do not have name-like identity, their *contents* reflecting dynamic properties of the world in these points. Forcefully changing the contents, local impact on the world can be done. PW nodes usually cease to exist if abandoned by all activities, or if aborted explicitly.

Virtual world. VW is universally represented by knowledge networks (KN) consisting of nodes and oriented or unoriented links connecting the nodes. Nodes have names, being also their contents, by which they may be referred globally. More than one node may have the same name, so accessing nodes by names may be ambiguous. Nodes, however, have unique *addresses* in the virtual space, which may be used for their quick access. Links have names too. Movement in VW may be done by direct hops by node names and addresses, or by propagation via links (using their names and orientation, if needed). KN nodes are persistent: after creation, they exist until deleted explicitly. Removal of nodes is accompanied by deletion of adjacent links. KN may have any topology, node and link names may by arbitrary long strings expressing any data, including programs in WAVE or any other language, to be executed.

PW & VW integration. PW nodes can be extended by properties of VW. For example, they can be made persistent, regardless of presence or absence of activities, may have unique addresses, and may also be interlinked. In their turn, VW nodes may be bound to physical locations and have coordinates for an access within PW. These coordinates may be forcefully changed, causing VW nodes move in space physically, while preserving existing links with other (stationary or mobile) nodes. Enriched PW nodes can exhibit a sort of identity and can move in space, being not considered as new ones at the destination. PW and VW nodes may also have any links between themselves.

Execution world. EW is usually hidden from user, but knowing and influencing its peculiarities may help optimize solutions. Any activity in both PW and VW is performed by *doers*, represented formally as nodes too, with unique both names and addresses. Doers support subsets of PW and VW nodes and their links, along with links to other subsets in other doers. Doers associate with PW coordinates, which change when they move, this movement being triggered by the creation of new PW nodes or movement of PW-bound VW nodes. Other doers may be accessed by their names or addresses, also locations in PW. Doers may have named links between themselves, as well as with PW and VW nodes. Addresses of the latter may consist of two parts: a unique doer address, and a unique position in

its memory. It is possible to refer to particular (or all) PW and VW nodes served by a particular doer, nodes may change doers at runtime. Node references may be two-stage, for example: "located in doer(s) situated in a certain region in space." A possible snapshot of this dynamic integral world, processed by the same universal space-conquering automaton, may look like the one in Fig. 2.

* temporary PW nodes   ▲ persistent PW nodes
○ free VW nodes   ● PW-boud VW nodes
☐ doers   —— named links
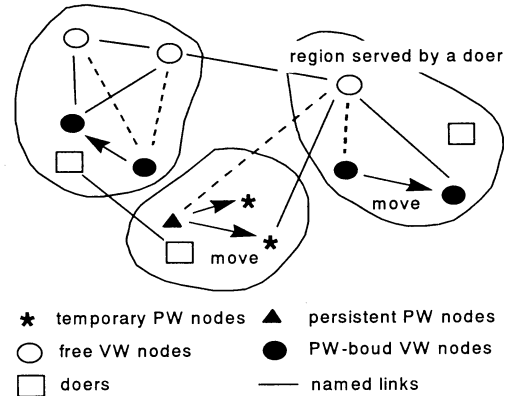
Fig. 2 United physical, virtual, and execution world

3 Extended WAVE Language

General organization. WAVE programs (or waves) evolve and act in distributed spaces regardless of their (physical, virtual or execution) nature. Waves start from any EW node, then create PW and VW nodes and propagate between them. Arbitrary actions can be performed in nodes, including lifting or changing node and link contents, and creating or removing other nodes and links. Additional, temporary data may be left at nodes and shared with other waves, while other data can move with waves. The data may represent both information and physical matter.

Many waves can evolve simultaneously in the same or different nodes, coordinated by distributed recursive control evolving in space together with waves. Staying in nodes, waves may search physical or virtual space to a given depth, and collect or change data in the nodes seen, interacting with other waves. WAVE has a recursive syntax, its top level shown below, where coordinated propagation in space can be integrated in the same expressions with collecting and processing of local and remote data.

```
wave     ->    { { move , } ; }
move     ->    value | [ rule ] ( wave ) | { move  act }
value    ->    constant | variable
variable ->    nodal | frontal | environmental
act      ->    control_act | fusion_act
rule     ->    explore_rule | echo_rule
```

Words in the syntax description are syntactic categories, braces show zero or more repetitions of a construct with a delimiter at the right, square brackets identify an optional construct, and a vertical bar separates alternatives.

Semicolon allows for a sequential, while comma for parallel (or arbitrary order) invocation of waves. Sequential parts of waves, or *zones*, develop independently from all nodes of the set of nodes reached (SNR) by a previous zone (regardless of the nodes nature), while parallel waves in a zone, or *moves*, all develop from the same current nodes, adding their SNR to that of the zone. SNR may contain nodes repeatedly, reflecting splitting & intersecting waves in the same locations, and subsequent SNRs may include nodes of the previous SNRs, allowing for loops in space.

Moves can be of three types. First, they can point at a resulting value directly (as a constant or variable). Second, they can themselves be arbitrary waves (in parentheses), optionally prefixed by control rules. Third, they can form space navigating & data processing expressions consisting of arbitrary moves separated by elementary operations, or acts, where moves may return their (including remote) results on a demand of acts, or assign the results to (including remote) variables.

Variables, acts, and rules. There are different types of distributed dynamic variables the spreading waves operate with. *Nodal*, or stationary, variables are created in nodes and may remain there, shared by different waves traversing the nodes. *Frontal* variables travel with waves, replicating when waves split. *Environmental* variables access the navigated spaces and execution layer. *Control acts* allow, direct or halt program and data flow through nodes where they are executed, they can also access other systems. *Fusion acts* provide data processing, they return results to be used by other fusion or control acts. A number of fusion acts can be applied to physical objects and their storage, while other operations may need external functions.

Rules establish a variety of constraints upon the development of waves. *Forward rules* coordinate spreading of waves in space. Among them, *branching rules* split the wave and coordinate parallel or sequential development of branches in space. Other forward rules include repetition of the spatial navigation, remote logical synchronization, protecting access to common resources, granting autonomy to waves, allowing the spreading waves create distributed networks, etc. *Echo rules* accumulate, generalize, and process (including remote) states or results reached by the embraced wave, returning them to the rule activation node for further assessment.

4 WAVE-Based Mobile Robot Architecture

Basic organization of the WAVE interpreter[2] designed for work in VW is transferable for the current extended VW-PW-EW applications. Strategies of runtime assignment of interpretive vehicles to waves for the PW case and dealing with shortage of them have been investigated,[3] with trade-off between the use of mobile hardware and mobile agents propagating between the hardware units. WAVE interpreter, substituting the von Neumann-type computer, may

efficiently integrate any other functional hardware and software modules of advanced robots.[6,7] Especially useful may be its symbiosis with specialized hardware *neurocomputers*,[7] which can provide very fast perception, pattern recognition, and local decision-making, leaving the top level planning, reasoning, and overall (both individual and group) control for the WAVE model organizing and processing distributed dynamic knowledge networks in a universal pattern-matching mode. A sketch of such a hybrid cooperative architecture is depicted in Fig. 3.
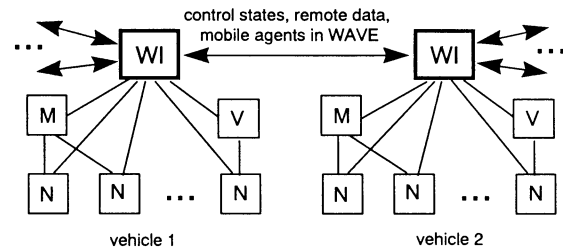
Fig. 3 WAVE-based mobile robot architecture: WI – communicating WAVE interpreters; M – motion/mechanics control; V – vision control; N – neurocomputers

5 Programming Scenarios

We will try to show here how easy may be the expression of some parallel and distributed scenarios in PW and its integration with VW in the extended WAVE.

Building a house. Let us consider a fragment for laying down a basement, where # is a hop act, and coordinates Xi and Yi of the locations of raw materials, together with speeds Si of reaching them, are represented as strings. Construction takes place on a plot with given coordinates. Physical matter is in double quotes, which means that its proper amount must be withdrawn from PW.

```
DIRECT # Xplot_Yplot;
((DIRECT # X1_Y1_S1; "10 tons of water") +
 (DIRECT # X2_Y2_S2; "8 tons of sand") +
 (DIRECT # X3_Y3_S3; "2 tons of cement"),
 (DIRECT # X4_Y4_S4; "50 tons of stones")
) ? lay_down_basement
```

Making mortar from mixture of sand, water and cement delivered from some remote locations uses an extended semantics of the act +, the result of which together with stones (also delivered from elsewhere) form together two parameters of the external procedure lay_down_basement, called by act ?. As can be seen, this PW scenario with natural parallelism of delivery of materials and construction actions is comparable on complexity to usual arithmetic expressions for processing numbers. The house may be completed in a similar way.

Map-based collection of samples. This scenario assumes that the planet surface must be simultaneously navigated by multiple rovers, all starting from a common

landing point. The rovers should collect soil samples in the points given, and return them all to the lander. The surface navigation is assisted by a territory map represented as KN, with nodes keeping the needed PW point coordinates, and links (marked with expected propagation speeds) reflecting available paths between the points, as the terrain may be complex, like the one shown in Fig. 4.

Fig. 4 Simplified territory map as a knowledge network

The map in Fig. 4 can be easily created and kept in a totally distributed form by the following parallel wave program, which is structured as a possible depth-first spanning tree template[2] of the KN, where act ##, in the context of CREATE rule, adds new links to the already existing nodes.

```
CREATE(
  DIRECT#60_45; 20#30_45; 15#55_65; 9##60_45,
  (5#125_70; (3#100_55; 4##55_65), (1#140_30;
  5#115_30; 3#65_20; 10##30_45, 8##60_45)))
```

The actual landing place in the region of interest may depend on different circumstances (say, weather), so after the landing a proper starting point on the map should be chosen, then optimal access routes to other points found. The following parallel program first finds the nearest point on the map to the landing, and creates shortest path tree to all other points using the distributed map. It subsequently navigates the surface physically, according to the tree found, by a recursive procedure Fbinding, gradually mapping the virtual tree nodes onto the corresponding PW locations. Samples are picked up in the points reached and returned to the landing site via optimal routes, being ultimately all stored in the PW-bound variable Nsamples.

```
SEQUENCE(
 (Flanding = (150,58); Ndistance = INFINITY;
  DIRECT # ANY; Fdistance =
      SQRT(SUM(Flanding -(CONTENT|_)**2));
  DIRECT # PREDECESSOR; Fdistance < Ndistance;
  Ndistance = Fdistance; Node = PREDECESSOR
 ),
 (DIRECT # Node;
  SEQUENCE(
   (Ndistance = 0;
    REPEAT(ANY # ANY; Fdistance +=
        SQRT(SUM((CONTENT|_)-
        ((DIRECT # PREDECESSOR)|_)**2))/LINK;
     Ndistance == NONE, Fdistance < Ndistance;
     Ndistance = Fdistance;
     Npredecessor = PREDECESSOR
    )
   ),
   (Fbinding =
    {WHERE = CONTENT; ? lift_samples,(ANY # ANY;
```

```
      Npredecessor == PREDECESSOR; ^ Fbinding)};
  Nsamples = (^ Fbinding)))))
```

The simplicity of this program can be explained by the fact that it describes semantics of the problem directly, without implementation details. Dynamic optimizing mapping of this high-level spatial scenario to available mobile hardware can be carried out in a fully formal and automatic way.

6 Conclusions

The world has become a huge networked computer. To rule it properly and utilize its distributed resources efficiently, radically new computational and control models are needed, oriented on solving global, spatial problems. Traditional computational models and technologies are making only first steps in this direction. If to compare them with organization of a traditional computer, they develop rather "functional units" and "registers" (say, agents and objects), and communication between these (message passing, RPC, object brokers, etc.), being far from assembling a workable network computer yet. Nor have they reached even the level of machine code to program such a computer as a whole.

WAVE aims at bridging this gap by offering high-level integral programming, in the form of active self-evolving spatial scenarios, of arbitrary tasks in networked hardware, which may have dynamic, loose, and open nature. Instead of a traditional sequence of commands over data stored in computer memory, it provides recursive navigation in distributed, both physical and virtual, worlds (PW and VW), within which they can be processed, modified, and created in a highly parallel manner. Traditional hardware and software techniques are used on the implementation layers only. This helps to relieve the programmer from many synchronization, coordination, and message passing details, and make application programs extremely simple and compact.

References

1. Hosmer DM (1999) The robotic fire fighting vehicle. In: Proceedings of AUVSI'99, Baltimore, MD, July 13-15, 1999
2. Sapaty PS (1999) Mobile Processing in Distributed and Open Environments. John Wiley & Sons, New York, 416p
3. Sapaty PS (1999) Cooperative conquest of distributed worlds in WAVE. In: Proceedings of AUVSI'99, Baltimore, MD, July 13-15, 1999
4. Minsky ML (1988) The society of mind. Simon & Schuster
5. Lange DB et al (1998) Programming and deploying Java mobile agents with aglets. Addison-Wesley, 288p
6. Sugisaka M, Wang X, Lee J-J (1997) Control strategy for an intelligent mobile vehicle. Artif Life Robotics 1:185-190
7. Sugusaka M, Tonoya N, Furuta T (1998) Neural networks for control in an artificial brain of a recognition and tracking system. Artif Life Robotics 2:119-122

Towards a common set of experimental frameworks for the evaluation and benchmarking of mobile robot control architectures.

Malachy Eaton, J J Collins, Lucia Sheehan
Dept. of Computer Science and Information Systems
College of Informatics and Electronics,
University of Limerick, Ireland.
Email: Malachy.Eaton@ul.ie, J.J.Collins@ul.ie, Lucia.Sheehan@ul.ie

Keywords: robotics, benchmarking, performance evaluation

Abstract

In this paper we suggest that one of the more crucial tasks currently facing researchers into the field of autonomous mobile robotics is the provision of a common task, or set of tasks, as a means of evaluating different approaches to robot design and architecture, and the generation of a common set of experimental frameworks to facilitate these different approaches.

This paper starts with a brief introduction to the field, and behaviour based control in particular, we then discuss the issue of animal versus robot behaviour. We then focus on the issue of simulated experimentation versus embodied robotics. We then move on to the area of the feasibility of the evaluation and the benchmarking of different architectures, with the aim of producing mobile robots of continuously higher utility, with specific reference to our current four layered robot control architecture.

Introduction

One of the basic tenets of traditional approaches to artificial intelligence has been the creation of internal *models* of the world, embodied intelligence, on the other hand, is focused on *action* in the world [1]. These are two quite different philosophical approaches to artificial or to computational intelligence.

Naturally intelligent systems must be understood in terms of the environment in which they operate and in terms of how they evolved to that state. This implies that their computational structures cannot be understood in isolation from, and indeed are directly intertwined with the mechanisms by which they perceive and alter their environment

In recent years much work has been done in the application of mobile robots - sometimes referred to as *mobots,* to a variety of tasks, from basic navigation through a known environment, to giving guided tours to people and in studies of pursuit-evasion strategies. [5] While many of these applications are interesting in their own right, perhaps less attention has been paid to the evaluation or benchmarking of different mobile robot performances to each other. This is not surprising, given the wide diversity of mobile robot architectures and assigned tasks. It might be compared to the difficulty in comparing the performance of two different, and very dissimilar, computer architectures.

However, if we are to be seen to construct progressively more useful robots it is essential to have some method of comparing relative performances. The situation is compounded somewhat by the fact that many researchers in the field are trying, consciously or otherwise, to mimic certain aspects of animal behaviour. This is, again, not surprising as many researchers would see the ultimate (although far distant) goal of their research as the production of a human-like robot - with human-like intelligence.[2]

This also poses the interesting question which was raised at a recent seminar on Bio-Robotics organised by the Institute of Electrical Engineers, by Prof. Alex Kacelnik of the University of Oxford, which is: what exactly are current researchers in this field trying to do anyway?

Is it

■ to build useful machines or,

- understand (and simulate) animal behaviour better?

This question is perhaps well typified by the work of Dr Hugo de Garis and his colleagues in the Brain Builder Group in ATR, Japan, who is trying to construct a robot 'kitten' with 10,000 separate modules controlling the kitten's many behaviours. This paper explores some of the issues posed by these vexing questions.

Behaviour based robotics

Many implementations of mobile robotic systems derive from 'traditional' AI (artificial intelligence) methodologies which rely heavily on the creation of sometimes very complex models of the external world. One first creates this model, then explores potential solutions to the task at hand using the model created, finally applying these 'solutions' to the real world through the robot's actuators. A problem here is that the middle two stages of the process described above - the model creation and planning processes, can prove to be extremely time consuming, thus producing a robot that may operate very slowly, say a move every 15 minutes, and may also be constrained to operate within a highly artificial constrained and artificial environment. Many of these systems also tend to operate as simulations rather than as 'real' or embodied robots.

Reacting to these perceived shortcomings, behaviour based robotics proposes turning the hierarchical control structure onto its side, having a number of behaviour generating modules operating asynchronously in parallel, each of which has access to sensory input and motor output. The proposal is that by starting with relatively few layers, e.g. move randomly, avoid obstacles, then by incrementally adding more layers robots of arbitrary complexity and sophistication may be constructed, even, in the case of the most ardent supporters of the approach, which include Rodney Brooks of MIT, up to and beyond the level of human-like competence.

Detractors of the approach claim that it does not scale well, as more behaviours are added, certainly not near the levels proposed by Brooks, and also point to the difficulty in defining individual behaviours. Consequently a lot of current work in the area involves testing the boundaries of behaviour based systems.

Many researchers nowadays espouse a hybrid deliberative- reactive approach [7] , drawing on what they see as the best aspects of both types of robot control: from purely symbolic computation to reflexive action, with the deliberative end taking care of complex sensory pre-processing, such as vision.

Two key components of the behaviour-based approach are situatedness and embodiment. By situatedness we mean that the robot deals directly with, and takes input from the real world, and not abstract descriptions; by embodiment, that the robot has a physical body, and is not just a computer simulation. Brooks [6] nicely distinguishes between these two concepts with the examples of an airline reservation system which is situated but not embodied, and a spray-paint robot which is embodied but not situated - it just goes through a pre-programmed set of actions.

A final concept is that of emergence - where systems must be understood in terms of the environment in which they operate and any intelligent behaviour displayed cannot be divorced from this environment. . This is a key feature of behaviour based robotics.

Definition of goals

In order to evaluate the performance of a particular architecture we must define clearly the goals of the robot. There is unease among some researchers in the mobile robot field as to the function of their research – is it to engineer useful products, or as an aid to the study and understanding of animal (and human) behaviours.

Gat [4] takes the approach of treating the experimental study of autonomous mobile robots as one would deal with a natural science such as biology. This is an interesting approach, which perhaps warrants further study.

We echo these concerns, and while it may be inspirational, and a certain background driving force to think that some day we may build the intellectual and physical equals of humans, for the present perhaps this is a question we should all ask at the start of a new research project in this field.

. For further discussion on this subject see [2].

Simulation versus Embodiment

Floreano and Mondado [9] point out also that given the variation in the backgrounds of researchers interested in the field of mobile robots, and evolutionary robotics in particular, including psychologists, biologists, and philosophers, very few of these will have the necessary skills to build their own robot, and to program it. This situation is changing gradually over the years, with more off-the - shelf robots suited to academic research becoming available (e.g. Khephera, Pioneer), but still remains an issue.

Miller[8], on the other hand, while broadly critical of research methods in the Artificial Life community in general views computer simulations as being of potential aid to practitioners of theoretical biology, particularly in the fields of new and innovative approaches to evolution, the interaction between different adaptive forces, the origins and the effects of adaptations in an organism, and in "life as it could be". Indeed he proposes that *"if successful,* A-Life will be absorbed entirely into evolutionary biology as a useful new modelling tool.".

His criticisms of the A-life field stem from several of his perceived shortcomings in the area:- a poor analysis of results in avoiding recognising failures and overgeneralising 'good' results; a failure to replicate and extend findings; ignorance of current work in the relevant sciences, and inadequate hypothesis-testing, systematic observation, controlled experimentation, and hypothesis testing.

While not dealing explicitly with autonomous mobile robot research, and given the relative infancy of the field compared to other biological and physical sciences, his words nevertheless present a sobering antidote to some of the hype presented in the media, and by a small number of practitioners of the field. Some researchers would also argue that research methodologies in the field have improved considerably since this paper was written.

Another set of questions arises: if both simulation and experimentation in the real world is performed (as seems to happen more commonly in recent years), we need to decide on the relative functions of these activities. Is the function of the embodied experiments to test the validity of the simulator produced, or is the simulator just a "thought-experiment" to be discarded once experimentation on the real robot begins? Is the policy (as is the case with some evolutionary roboticists) that the simulation and embodied experiments are performed hand-in-hand, with results from one activity forming input to the other? This policy corresponds broadly with Brooks' suggestion of interleaving simulation and embodied experimentation as a means of "connecting to reality".[10].

Clearly there are divided views on the subject. Given the variety of views presented, it is suggested that perhaps some common framework could be found, giving guidelines for the different strands of mobile robotics research, as to the usage and interaction of simulated and embodied experimentation, to facilitate the comparison of different architectures.

Benchmarking

A current *de facto* standard in this field is Robocup annual challenge. This has it's origins in the 1992 Dynamo project in Canada, leading to the first Robocup tournament in 1997, and the third tournament in 1999 in which a total of 95 teams participated. Robocup operates in four categories: simulated teams, a small size league, a middle size league, and legged robots. An example small size robot is Khephera, a typical middle sized robot is Pioneer (1/2) and The Sony artificial dog would fit in the third category. There is also a humanoid league. Robocup tests skills at the individual, the inter-individual, the collective and at the competition level.

Individual skills to be mastered include navigation and localisation on the field of play, including the selection of optimal paths. Inter-individual skills include the coordination of movements with playing partners in order to pass accurately. At the top level the tasks of strategy generation and recognition of opponents' strategies are crucial. [3]

Criticisms of Robocup stem from the controlled environment in which the robots operate, and the fact that soccer playing skills are quite specific and may lead to the development of highly focused robots of little use for any other task.[3] There is also the thorny issue of credit assignment. Given the complexity of the task facing the robots, both

individually and collectively, and the relative paucity of the ultimate reinforcement signal; i.e. scoring of a goal, how do we decide which action(s) to reward and reinforce.

So while Robocup may currently be a useful testing bed for approaches to Artificial Intelligence and Artificial Life, problems exist. One potential approach, which we espouse, involves the provision of a set of specifically designed experimental frameworks, based loosely on the Robocup model involving tasks of increasing complexity, rigorously defined to facilitate experimental reproducibility and verification.

Four layer model

Initial work in this area centres around a four layer model for pursuit-navigation, outlined below. Space precludes a detailed discussion of this model, however further details are presented elsewhere in these proceedings. [11]

Level 1: Reactive Control

Given: A global map of the environment. This environment has obstacles not shown on the global map: e.g. doors closed, boxes in corridor.
            A global navigation strategy.

Task: Reach target avoiding obstacles in the way.

Level 2: Navigation to a goal in a static environment

Given: A global map of the static environment.
Task: Navigate to target, minimizing one, or a combination of factors:- e.g. time taken, distance   travelled.

Level 3: Pursuit- evasion.

Given: A global map of the environment.
            A global navigation strategy.
            A reactive control algorithm.

Task.   Seek out, pursue, and catch an evader, learning evasion strategies over time, and modifying   pursuit strategies over time in tandem.

Conclusions

One of the more important tasks currently facing researchers in the fields of artificial life and embodied intelligence is the provision of common benchmarks for performance evaluation. Current benchmarks, while useful, have their problems.   We advocate the generation, from the bottom up, the generation of a common set of experimental frameworks, for performance evaluation and evaluation. Our current research is taking tentative steps in this direction.

References

[1]Cliff   D(1998),   6.386   Embodied Intelligence, Course Notes, MIT AI Lab, 1998.
[2]Collins J J, Eaton M(1999). " Open issues in the design and synthesis of intelligent robot control architectures" *Proceedings of the 10th Irish Conference on Artificial Intelligence and Cognitive Science*, pp. 108-116.
[3]Drogoul Alexis(1999) Keynote speech at *10th Irish Conference on Artificial Intelligence and Cognitive Science*
[4]Gat Erann (1995)"Towards a principled experimental study of autonomous mobile robots", *Autonomous Robots*, Vol. 2, 1995 pp. 179-189.
[5] Collins, J.J. and Eaton, M. (1998). "Situated Pursuit and Evasion Using Temporal Difference Learning".*2nd Int. Conf. on Cognitive and Neural Systems*, May 1998, Boston, MA, p.100
[6]Brooks   R(1991),"New   approaches   to robotics", *Science*, Vol. 253   Sept. 1991 pp.1227 - 1232
[7]Arkin R(1998),"Behaviour based robotics", MIT press, 1998.
[8]Miller   G,F(1995)"Artificial   life   as theoretical biology: how to do real science with computer simulation", University of Sussex, CSRP 378.
[9]Floreano D, Mondada F(1998)"Hardware solutions for evolutionary robotics" in *Evolutionary Robotics, Proceedings of the first European workshop, Springer Verlag LNCS 1468*, pp.137-151.
[10]Brooks R(1992)"Artificial life and real robots" *in "First European conference on artificial life"*, MIT Press, pp. 3-10.
[11]Collins J J et al(1999)"Four layered robot control architecture" Proceedings of the Fifth International Symposium on Artificial Life and Robotics, 1999.

An Autonomous Restoration of Efficiency for AGV Transportation System

Toshimitsu HIGASHI†, Kosuke SEKIYAMA‡, and Toshio FUKUDA*

†Automated Systems Division, Murata Machinery, LTD.

2, Nakajima, Hashizume, Inuyama, Aichi 484-8502, JAPAN

‡Dept. of Management and System Science, The Science University of Tokyo, Suwa College

5000-1 Toyohira, Chino, Nagano 391-0292 JAPAN

*Center for Cooperative Research in Advanced Science & Technology, Nagoya University

Furo-cho 1, Chikusa-ku, Nagoya 464-0814, JAPAN

higashi@robo.mein.nagoya-u.ac.jp, sekiyama@ss.suwa.sut.ac.jp, fukuda@mein.nagoya-u.ac.jp

Abstract

This paper proposes a system, which manages both efficiency and flexibility in the AGV(Auto Guided Vehicle) transportation system. In this paper, to realize the behavior so as to restore the system performance on managing efficiency and flexibility, we propose the method, where each AGV organizes the number of the all AGV in the system by self-organizing behavior through interaction in the local domain under the change of environment. It is shown that the system exhibits autonomous restoration of the system performance and higher efficiency compared with the centralized system by introducing self-organizing in the dynamical environment.

1 Introduction

This paper proposes a design technique for AGV(Auto Guided Vehicle) transportation system. The distributed autonomous design technique uses the self-organizing behavior of each agent to achieve the purpose of the system, high efficiency and flexibility. Flexible manufacturing system and logistic system are generally large, inflexible and vulnerable to change of specification which is the root cause of trouble in the long run. Therefore, we propose the distributed autonomous design technique be applied to those systems. Through this approach, the design technique realizes management of superior adaptability and high efficiency in the dynamic environment as depicted in figure 1, compared with centralized system design approaches. On the multi-agent system, Fukuda et al. describe and perform a research program about dy-
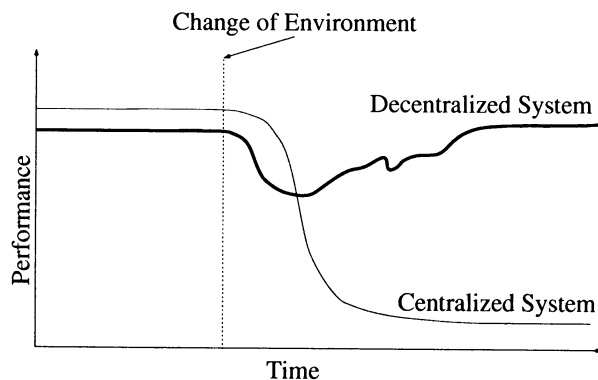
Figure 1: Restoration Behavior in Distributed System

namically reconfigurable robotic system[1]. We have worked on the distributed autonomous system which realizes management of superior adaptability and high efficiency in the AGV transportation system in the dynamic environment, without using the global evaluation function[2]. However, the essential feature of self-restoration behavior in the system, as shown in fig 1, could not be realized in previous work. Therefore, in this paper, we attempt to simulate this technique to verify if it is possible to restore the transport efficiency through the self-referential interaction of agents within the local domain. Where, the self-referential interaction, which is a key concept in this paper, is referred as a recursive process described as follow; the communication process of agents concerning evaluation value generates a strategic collective behavior, and the same time, the self-organized strategy formation sustains the interactive behavior of individual agent. Further, to understand the phenomenon of autonomous restration of transport efficiency, we examine a change of
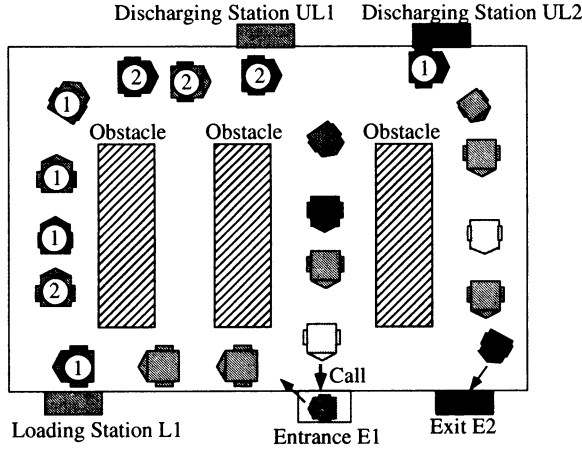
Figure 2: Self-organizing AGV transportation system

strategy formation with self-referential interaction and test its validity.

2 The Model of Research

2.1 The outline of the model

Figure 2 shows the proposed model for research. Model selected for this research is called 'Self-organizing AGV transportation system', consisting of several numbers of AGVs (later each termed as 'carrier i,e agent). Each carrier carries the baggage from $L1$, a loading station, to $UL1$ or $UL2$, a discharging station, autonomously. The carrier can select one discharging station of two to put a piece of baggage. Two kinds of baggage are assumed and they are labeled as $\{1,2\}$ at random. The purpose of the model is as follows: (1)To archieve highest transport efficiency per carrier. (2)To transport baggage to each discharging station equally. In the system, carrier cannot have the global map of the system. The number of working carrier in the system is self-regulated according to the work load, where the detailed method follows the previous work[2]. About communicating with each carrier, the carrier usually cannot obtain the information of all other carriers but only in the sensing range, and it cannot select the label of the baggage.

2.2 The method of selecting task

We explain the system configuration and the procedure of task selection of the agent. The number of the carrier in the field of the system is shown with eq.(1),

and it is composed of the number of the carrier which selects tasks from S_1 to S_5.

1. TaskS_1 : Transport a baggage to $UL1$.

2. TaskS_2 : Transport a baggage to $UL2$.

3. TaskS_3 : Move to $E1$, Call another carrier, (and Copy stochastic strategy vector).

4. TaskS_4 : Exit the field of the system through $E2$.

5. TaskS_5 : Move to $L1$, and load a baggage. (the task is selected at only $UL1$ and $UL2$)

$$N(t) = \sum_i N_i(t) \quad \{i = 1, 2, 3, 4, 5\} \tag{1}$$

Where, $N_i(t)$ is the number of the carrier which selects the task S_i.

Each carrier is required to transport baggages to each discharging station equally without general evaluation function, and it must maintain high transport efficiency. Therefore, we define the task S_1 and S_2 as determining discharging station, and the task S_3 and S_4 as maintaining high transport efficiency. We use the transport value $E(t)$ and the design value of transport $E^*(t)$ as the parameter for estimating the system performance. $E(t)$ is amount of transport (the sum of $UL1$ and $UL2$) at the time t, and $E^*(t)$ is desired value of transport value $E(t)$ at the time t.

Desired value of transport $E^*(t)$ is found with $E_i(t)(i = 1, 2)$. $E_i(t)(i = 1, 2)$ are parameters for amount of transport to the site $UL1$ and $UL2$ respectively.

$$E^*(t) = \frac{1}{2} \sum_{i=1}^{2} E_i(t) \tag{2}$$

Learning method of the carrier is explained as follows. The carrier learns the task S_1 and S_2 which determines discharging station. We use the parameter r_i as reward, and the carrier learns its behavior of the strategy so that $E(t)$ may approach $E^*(t)$. The carrier learns at the site of $UL1$ and $UL2$ concerned with the situation of i) and ii), when the carrier discharges a piece of baggage as shown with eq.(4) and eq.(5). By definition a difference of $E(t)$ and $E^*(t)$ as $\Delta E(t)$,

$$\Delta E(t) = E(t) - E^*(t) \tag{3}$$

i)The case of succeeding in discharging:

$$r_i = \begin{cases} -r_c & \text{if } eE^*(t) \leq \Delta E(t) \\ -\dfrac{2r_c}{eE^*(t)} \Delta E(t) + r_c & \text{if } -eE^*(t) \leq \Delta E(t) < eE^*(t) \\ r_c & \text{if } \Delta E(t) < -eE^*(t) \end{cases} \tag{4}$$

ii)The case of failing in discharging:

$$
r_i = \begin{cases} -r_c & \text{if } eE^*(t) \le \Delta E(t) \\ -\dfrac{r_c}{eE^*(t)}\Delta E(t) & \text{if } -eE^*(t) \le \Delta E(t) < eE^*(t) \\ 0 & \text{if } \Delta E(t) < -eE^*(t) \end{cases} \tag{5}
$$

Where, r_c is the regulated value of reward r_i, and e is the allowable error of amount of transport.

The task selecting probability of S_1 and S_2 are obtained with eq.(6), by using these parameters, amount of transport $E(t)$, desired value of amount transport $E^*(t)$, and the reward r_i.

$$
\rho_{cS_i wj}(t) = \rho_{cS_i wj}(t-1) + r_i(E_i(t), E_i^*(t)) \tag{6}
$$

$$
\rho_{sS_i wj}(0) = 0.5 \tag{7}
$$

Where, i is the number of task S_i $(i = 1, 2)$, and j is the number of the baggage w_j $(j = 1, 2, 3, 4)$.

S_3 and S_4 are the task which maintain high transport efficiency, and are obtained by $R_c(t)$ and $I_c(t)$. $R_c(t)$ and $I_c(t)$ are functions which is described about the interaction with other carriers. Details of these functions are explained at section 2.3.

The formula which founds stochastic strategy vector V_c is shown with eq.(8) and eq.(9).

$$
V_c' = \begin{bmatrix} J(S_1) \\ J(S_2) \\ J(S_3) \\ J(S_4) \end{bmatrix} = \begin{bmatrix} \rho_{cS_1 w_1}(t) & \rho_{cS_1 w_2} & 0 \\ \rho_{cS_2 w_1}(t) & \rho_{cS_2 w_2} & 0 \\ R_c(t) & R_c(t) & R_c(t) \\ I_c(t) & I_c(t) & I_c(t) \end{bmatrix} \begin{bmatrix} w1 \\ w2 \\ w3 \end{bmatrix} \tag{8}
$$

$$
V_c = \begin{bmatrix} p(S_1) \\ p(S_2) \\ p(S_3) \\ p(S_4) \end{bmatrix} = \frac{V_c'}{\|V_c'\|} \tag{9}
$$

With the stochastic strategy vector V_c in eq.(9), each carrier selects the task, moves to the site concerned with the task, and processes the task.

2.3 The interaction in local domain

Task S_3 and S_4 are determined by the interaction with other carriers in its sensing range. The function about the interaction with the others $R_c(t)$ and $I_c(t)$ are obtained by eq.(10) and eq.(11).

$$
R_c(t) = \gamma \frac{(Q_c(t) - Q_d(t))Q_c(t)}{m} \tag{10}
$$

$$
I_c(t) = l_c(t) + \gamma \frac{(Q_d(t) - Q_c(t))Q_d(t)}{m} \tag{11}
$$

$$
l_c(t) = \begin{cases} l_c(t-1) - \alpha & \text{if } AGVc \text{ has a baggage} \\ l_c(t-1) - \beta & \text{if } AGVc \text{ does not have a baggage} \end{cases} \tag{12}
$$

Where, γ is potential evaluation parameter (fixed), l_c is rest of life for the carrier at the time t, α is life consumption rate (loaded the baggage), and β is also life consumption rate (unloaded the baggage).

The function $Q_c(t)$ and $Q_d(t)$ in eq.(10) and eq.(11), are parameters of the interaction with other carriers which is used for determining the task S_3 and S_4. $Q_c(t)$ is transport achievement within the unit time T of the carrier c, and it is defined as transort potential shown with eq.(13).

$$
Q_c(t) = \int_{t-T}^{t} (E_{c1}(\tau) + E_{c2}(\tau))d\tau \tag{13}
$$

Where, $E_{ci}(t)$ is amount of transport of ULi of the carrier c $(i = 1, 2)$, and T is the unit time of evaluating amount of transport.

$Q_d(t)$ is the average of whole transport achievement (sum of $UL1$ and $UL2$) for other carriers in the sensing range within the unit time T.

3 The definition of the environmental change

In this paper, we define the environmental change as the change of discharging rule at the site $UL1$ and $UL2$. transport is performed in the system under the environmental change as follows. While the step t is $0 \le t < 40000$, discharging stations allow all carriers to unload any baggage of any label. And while the step t is $t > 40000$, discharging stations allow discharging only if well the number of the baggage that the required label of the baggage. The simulation is worked up to 80000 step.

4 Simulation Result

4.1 Autonomous efficiency restoration

The evaluation of transport efficiency is shown in figure 3 on the condition described in section 3. Figure 3 is a dynamics of the transport efficiency $q(t)$ which is carrierd out per 100 steps. "Whole" in figure 3 is the total amount of transport, which is the sum of each task's transport $S_i(i = 1, 2, 3, 4)$. "S_1" and "S_2" in figure 3 are transport efficiency of the task $S_i(i = 1, 2)$ which are amount of transport of each discharging station divided by the number of the carrier
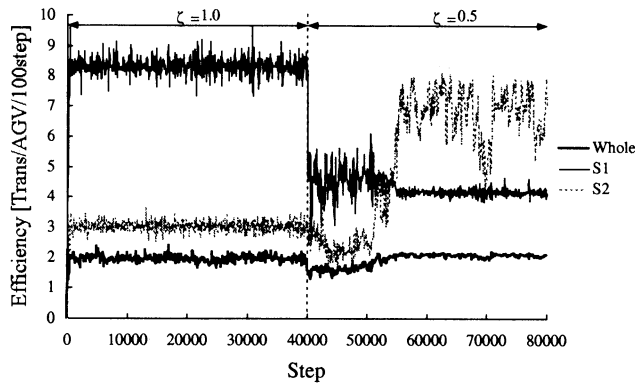
Figure 3: Evaluation of System Performance in the Dynamical Environment

Figure 4: Dynamics of Strategy S1 and S2 on Parameter Space

selected the task $S_i (i = 1, 2)$ respectively. As shown in figure 3, before the environmental change which is up to 40000 step, transport efficiency of "Whole" converges at about 2.0 baggage per carrier. It seems that 2.0 baggage per carrier is the optimized efficiency in this model because the success probability of discharging ζ is 1.0, and every carrier can discharge a piece of baggage every discharging station. At the time of 40000 step, the environment is changed so that ζ is 0.5, the transport efficiency of "Whole" sharply decreases by about 1.5. But through 48000 step to 55000 step, the transport efficiency of "Whole" is covered, and after 58000 step, it seems to be restored as well as the situation before the environmental change. Consequently, in this paper, it is considered that autonomous strategy reconfiguration becomes effective for managing high efficiency and restoration of the transport by using the interaction of each carrier.

4.2 Time evolution of autonomous strategy reconfiguration

Figure 4 explains about the strategy reconfiguration which satisfies autonomous restoration of transport efficiency. Figure 4 depicts the parameter space which basis is S_1-S_2, and shows a ratio of the number of agent selecting task S_1 and S_2 at the time step t. (a) and (b) in figure 4 are the rate of strategy formation of the carrier before the environmental change, (c) and (d) are that after the environmental change. It is found that the strategy formation is once converged by (a) and (b). But when the carrier faces the environmental change which is the change of success probability of discharging ζ, the carrier cannot maintain the strategy formation. So the strategy formation is diffused as shown with (c) in figure 4. As
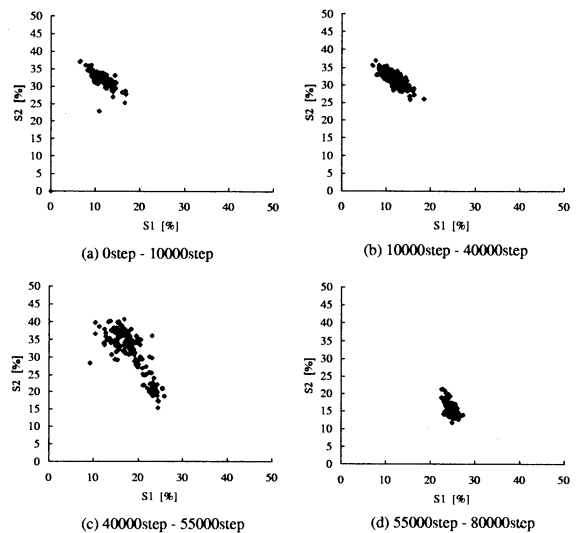
the component of the task S_1 and S_2 diffused at the environmental change, the carrier reforms the strategy which maintains high transport efficiency, at the end of the simulation, the strategy formation is converged again as shown with (d) in figure 4. Therefore, we can define S_1 and S_2 as the task which makes the strategy formation diffused, S_3 and S_4 as the task which makes the strategy formation converged.

5 Conclusion

In this paper, it is realized that the proposed system can maintain and restore the transport efficiency in the dynamic environment by the interaction within the local domain. In long term, the proposed system is possible that the efficiency is compatible with the flexibility by self-organizing behavior of the agent, even under the change of discharging rule drastically.

References

[1] Fukuda, T. and Ueyama, T. *Cellular Robotics and Micro Robotic System.* World Scientific in Robotics and Atuomated Systems. World Scientific.

[2] Higashi, T., Sekiyama, K., and Fukuda, T. Autonomous strategy organization in the dynamical quantitative environment. In *Proceedings of the 1998 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems,* pages 1547–1552, 1998.

Self-Localization of a Mobile Robot by Recognizing Color Objects in an Omnidirectional Image

Tae Kyun Kim   Young Jin Lee   Myung Jin Chung

Dept. of Electrical Engineering

Korea Advanced Institute of Science and Technology (KAIST)

ktk@cheonji.kaist.ac.kr

Abstract - For the autonomy of a mobile robot it is needed to know the position and orientation of a robot. Various methods of estimating the position of a robot have been developed. However, it is still difficult to localize the robot without any initial information. In this paper we present how to make the colored map and calculate the position and direction of a robot comparing the map with an omnidirectional image. We render the wall of the map with the corresponding color image and store the coordinates and color histograms of objects. Then a mobile robot gets the color omnidirectional image at random position with unknown orientation, segments it and recognizes the objects using the color histograms and spatial order. From the information of recognized objects, the robot can have reliable feature points and localize itself.

Key Words - self-localization, omni-directional vision, color histogram, image understanding

1. Introduction

Existing localization methods can be classified into following two groups; one is measuring the relative position using an encoder or a gyro, and the other is measuring the absolute position using a map. The relative method has disadvantage of accumulating error and can't estimate the position when it loses past data by an incident. Generally both methods are used together to compensate this problem.

In case of estimating the absolute position the reliability greatly depends on the similarity of a working environment and the accuracy of sensors. If the inputs of sensors are same about many positions, it will be impossible to know current location. To build a map which has unique data about every position it is very important to use various kinds of data and compare them effectively.

For the purpose of getting more information of the surroundings there have been studied various methods; sensor fusion method using vision and ultrasonic, robust method to camera movement using cross ratio

of feature points[1] and methods using angle data of feature points in an omnidirectional sensor[2] and etc. These methods provide correct results within an allowable error bound when the initial position and orientation are given. But considering the aspect of using the initial information they may have the same problem as the relative methods. They only used geometrical information of a map and using one kind of information it is difficult to localize a robot without any initial information in a complex and symmetrical indoor environment.

Recently there have been developed many methods using more various kinds of information based on image. Zhang used the parallel edge information and Matsumoto[2] used the intensity of image sequence and Schmitt[3] used texture data of an image and Cassinis[4] used total color distribution of an omnidirectional image.

In this paper we propose a method of estimating the absolute position and orientation of a robot in the map using the color omnidirectional image sensor without knowing the initial position and orientation. Color information is indispensable considering the geometrical symmetry of indoor corridor. The color information of objects can be effectively represented using color histograms and objects can be recognized by histogram matching. Based on the object recognition, the robot can get reliable feature points. Proposed algorithm is also robust to rotation of the robot by using the omnidirectional sensor. The block diagram of the proposed algorithm is shown below.
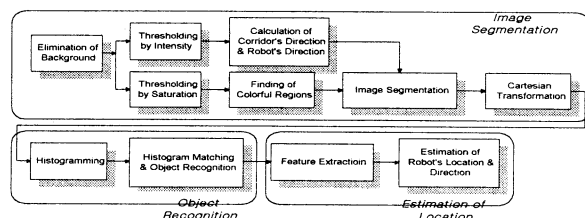
Fig. 1. System Block Diagram

2. Omnidirectional vision system and Selection of interesting regions

2.1 Omnidirectional vision system

Omni-directional vision system is able to get all directional view at one time without any particular movement of a camera. In this research we set the face of a conical mirror upward and put a CCD camera above it aligning the axis of the camera and the axis of the mirror as shown in the Fig.2. Especially parallel lines to the axis of the camera appear on the omni-directional image radiating from the center.
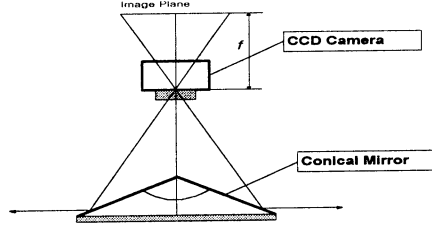
Fig. 2. Omni directional vision system

2.2 Selection of interesting regions

For the localization of a robot the interesting regions are color objects such as doors, boards and panels attached on the walls. First of all the image is segmented like Fig.3 using information of lighting source and HSI(Hue, Saturation, Intensity) color model. HSI color model is not largely changed by different ambient illuminations.
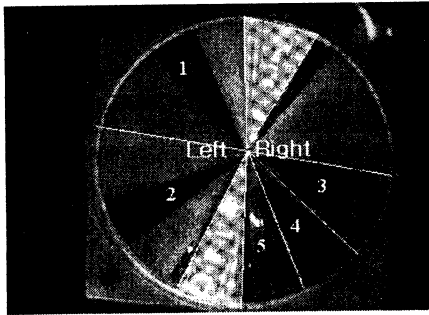
Fig. 3. Result of image segmentation

2.2.1 Elimination of background

Average hue and saturation of all pixels which have the same distance from the center of an image is selected as the color of background.

$$H_{bg} = \frac{\sum_{\theta=0}^{360} H(RGB(\rho\cos\theta, \rho\sin\theta))}{360}, \quad S_{bg} = \frac{\sum_{\theta=0}^{360} S(RGB(\rho\cos\theta, \rho\sin\theta))}{360}$$

We eliminate the pixels such that

$$\frac{H}{H_{bg}} \leq Th_1 \quad \text{or} \quad \frac{H}{H_{br}} \geq Th_2 \quad \text{and} \quad \frac{S}{S_{bg}} \leq Th_3$$

Because the saturation of background is generally

low, it can be eliminated effectively using above equations.

2.2.2 Selection of wall region using lighting sources

For all pixels which have high intensity, the direction is calculated and the most frequent direction is selected as the direction of corridor using the histogram voting. By eliminating neighbored regions to the direction of the corridor the robot can get left and right wall regions.

2.2.3 Segmentation of interesting regions

After the background is eliminated and the wall regions are selected at previous steps, remaining works are grouping the interesting regions on the wall and recognizing them. It is possible to segment the regions using the sequential labeling algorithm(8-connectivity). Small regions are eliminated and neighbored regions are combined. As a result, the robot can have fan-shaped partial images as shown in Fig.3. Polar coordinates is transformed into Cartesian coordinates and final images for recognition are obtained as follows.
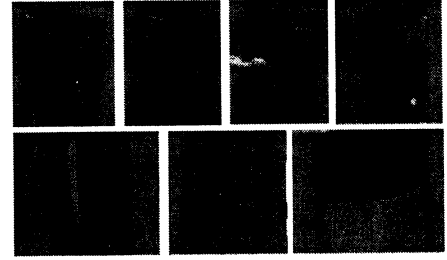
Fig. 4. Partial images after Cartesian transformation

3. Object recognition

When the shapes of objects are similar and the number is limited like Fig.4, color information is more efficient than geometrical information for object recognition. We first group the objects using both 2D histogram of hue and saturation and spatial sequence of the objects. After that, each object included in the group is recognized in order.

3.1 The color histogram group

HS histogram is changeable according to ambient light. For the robust recognition of objects we intersect all histograms of model objects and measure the similarity and classify the objects by the similarity. We referred to the Swain's paper[5] about color histogram and histogram intersection.

Table 1. Result of color grouping

Color group	5	4	3	2	1
1	0.196	0.247	0.36	0.24	0.99
2	0.505✓	0.241	0.317	0.99	
3	0.213	0.297	0.99		
4	0.282	0.99			
5	0.99				
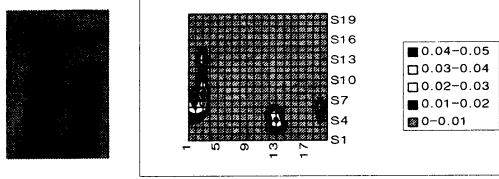
Fig. 5. HS histogram of color object

3.2 Map building based on color image

The 3D color map is made by following off-line procedure.

1. Get all images including color objects on both walls.
2. Select the objects which have large size and high saturation as landmarks.
3. Store the absolute position(L1,L2,...Lm) and the color histograms of the objects.
4. Group the objects according to the wall(M1,M2) and color distribution(C1,C2,...Cn).

3.3 Object recognition

Through the image processing explained in section 2, the objects are found and divided into two groups(S1,S2) about left and right walls. Then color histograms and angle data($\theta 1$, $\theta 2$,..., θk) of the objects in omnidirectional image are stored. Here, the robot can recognize the objects using the proposed interpretation tree algorithm under the following two assumptions.

Assumption 1: Color distribution of objects, which are not landmarks, is not similar to landmarks.
Assumption 2: Multiple objects in one omni directional image are within small distance.

1. Consider two possible cases one by one. {(S1,M1),(S2,M2)}, {(S1,M2),(S2,M1)}
2. Examine all combinations of objects in M1 and select the groups which have the same color group as S1 in spatial order.
3. Calculate the difference of the distance of the objects and select the groups with small difference.
4. Repeat Steps 2, 3 about the case(S2,M2).
5. For two cases{(S1,M1), (S2,M2)}, calculate the

difference of average distance of groups and select the pairs of group with small difference.
6. In case that no group is selected, repeat Steps of 2~5 about the case{(S1,M2), (S2,M1)}.
7. In case that single group is selected, recognize each object of the group using angle and location data.
8. In case that more than two groups are selected, select the most probable group using color histogram directly instead of the color group.
9. In case that the reliability of the group matching is low and the number of feature points are not enough, move the robot along corridor using the pre-calculated lighting source's direction.

4. Calculation of the position and orientation of a robot

Using the angles($\theta_0^R, \theta_1^R, \cdots \theta_N^R$) of landmarks in omnidirectional image and the location($L_0^W, L_1^W, \cdots L_N^W$) of landmarks in world coordinates, the position and orientation(R^W, θ_R^W) of a robot can be calculated.
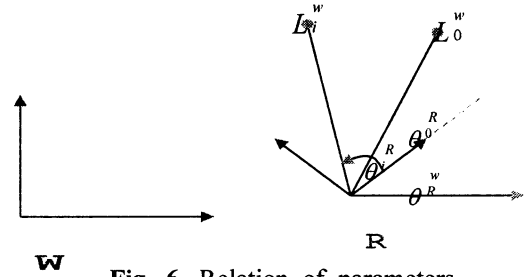
Fig. 6. Relation of parameters

We get eq.(1) using the relation of three 2D vectors and get eq.(2) substituting L_0-R and L_1-R for a and b.

$$\frac{1}{a} = \left(\frac{|b|}{|a|} e^{j\phi} - 1 \right) \frac{1}{c} \quad (1)$$

$$\frac{1}{L_0 - R} = \left(\frac{|L_1 - R|}{|L_0 - R|} e^{j\phi} - 1 \right) \frac{1}{L_1 - L_0} \quad (2)$$

Since one additional equation like eq.(2) is needed to find R, we need at least three landmarks[6]. To get sufficient feature points, we use two side edges of rectangular shaped objects like doors and bulletin boards. For reliability we use four landmarks which have the maximum average angles between two of them.

5. Experimental result

To evaluate the proposed algorithm, experiment has been carried out for corridor of the second floor of our department. The constructed 3D color map is shown in Fig.7. The number of selected feature objects is 24 and the number of the color groups is 5.
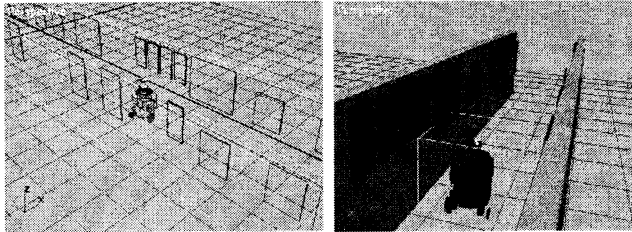
Fig. 7. 3D color map

Fig. 8 is the result of object recognition. The lower left rectangular represents the color of eliminated background. Lighting sources are blue-colored and small white squares are put on the recognized objects.
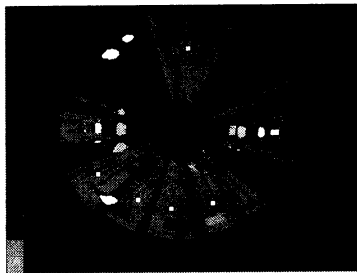
Fig. 8. Result of object recognition

Fig. 9 is the estimation result of the position and orientation when the robot rotates at specific points.

	average error
x	0.1642 (m)
y	0.1608 (m)
angle	5.29 (degree)
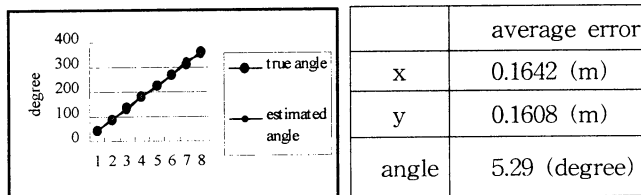
Fig. 9. Estimation result of position and orientation

Fig. 10 is the result of the localization when the robot is at arbitrary positions of the entire map.
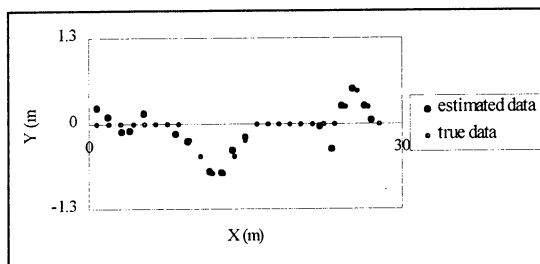
Fig. 10. (a) Result of position estimation
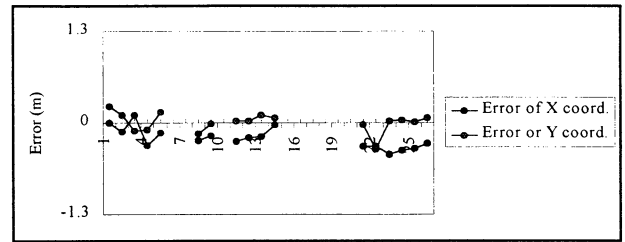
Fig. 10. (b) Error of position estimation

Execution time of whole algorithm for localization took about 850ms for Pentium II 400 PC. The maximum error of estimation for x axis is 1.3 percent and maximum error for y axis is 8.3 percent comparing with the dimension of the map. These values are quite good for navigation and the accumulating error of relative sensor can be compensated with these results. Data of unmeasurable positions due to similarity are not included. Error of data are due to various reasons; recognition error owing to change of illumination, measuring error of angles in omni-directional image and modeling error of the map.

6. Conclusion

We proposed a new algorithm for global positioning and environmental understanding. Using omni-sensor and color histogram, a mobile robot can recognize color objects and get more reliable feature points and localize itself in the real corridor without any initial data. The proposed algorithm can be easily applied to not only corridor but also other colored environments.

References

[1] K. S. Roh, "Obstacle detection and self localization without camera calibration using projective invariants," IEEE IROS '97.
[2] Y. Matsumoto, "Visual navigation using omni view sequence," Proc. of International Conference on Intelligent Robots and Systems, '99.
[3] Schmitt, M., " Vision-based self-Localization of a mobile robot using a virtual environment," International Conference on Robotics and Automation, '99.
[4] R. Cassinis, "Using colour information in an omni directional perception system for autonomous robot localization," Proceedings of EUROBOT '96.
[5] M. J. Swain, "Color Indexing," International Conference of Computer Vision, '91.
[6] M. J. Chung, "Development of intelligent robot and monitoring system in hazardous environment", Technical report published in KAERI '98.

On-Line Scheduling of Multiple Mobile Robot System
Using Coordination Protocol

Dong Hyun Yoo   Gil Whoan Chu   Myung Jin Chung

Dept. of Electrical Engineering

Korea Advanced Institute of Science and Technology (KAIST)

ydh@cheonji.kaist.ac.kr chgh@cheonji.kaist.ac.kr, mjchung@ee.kaist.ac.kr

Abstract – In these days, the control and the management of multiple mobile robot system are requested on many service fields such as office and hospital. In this case, the multi-robot system has to serve various demands of user successfully and efficiently which occur concurrently, irregularly, and independently. Multiple robots have more advantages than single robot. But, there are task assignment problem and scheduling problem in multiple robot system. In this paper, we suggest a protocol between robots and stations for on-line and decentralized scheduling which improves a performance of multiple mobile robot system. The proposed protocol considers robot-initiated task assignment rules and workcenter-initiated task assignment rules simultaneously. And, it is modeled in Petri net in order to check resource conflict and deadlock problems. The results of computer simulation show the feasibility and efficiency of the proposed protocol and on-line scheduling method.

Keywords – multiple mobile robot system, on-line scheduling, coordination protocol, DEDS, Petri net

1. Introduction

In these days, the control and the management of multiple mobile robot system are requested on many service fields such as office and hospital. In this case, the multi-robot system has to serve various demands of user successfully and efficiently which occur concurrently, irregularly, and independently. Thus, multiple robots take less time than a single robot because many robots process multiple tasks simultaneously. And another advantage of multi-robot system is reliability. If one robot is out of order, other robots can substitute it to achieve the given task successfully.

When the multi-robot system processes many jobs, how to assign tasks and how to schedule multi-robots significantly influence on the efficiency of system. These problems are called task assignment problem and robot scheduling problem.

The conventional scheduling methods are classified into two schemes; the robot initiated task assignment rule and the station initiated task assignment rule. Because those methods are unilateral scheduling methods, they could not function effectively in dynamic environments. In this paper, we present an on-line and

decentralized scheduling method that improves performance of multiple mobile robot system. On-line scheduling is suitable for the dynamic environments where we cannot predict when the request occurs. In order to act without stopping whole robot system when one or more robots are out of order, the scheduling of multi-robot system must be decentralized. For on-line and decentralized scheduling, we suggest a coordination protocol. This protocol is a communication scheme between robots and stations. It considers robot initiated task assignment rule and station initiated task assignment rule simultaneously. The protocol is based on DEDS(Discrete Event Dynamic System) theory. When the schedule is determined by the proposed method in the multiple mobile robot system, resource conflict and deadlock can occur. In order to check these problems, we model the robots and stations using Petri net. Petri net is a suitable modeling tool to evaluate the performance of DEDS. We use Predicate/Transition net that is an extended version of Petri net because Predicate /Transition net can represent a variety of real systems.

Performance indices are defined to evaluate the performance of multiple mobile robot system. Those indices are the mean wait time and the mean tardiness. In order to confirm the feasibility of the proposed protocol, the computer simulation is carried out. The simulation results are compared with conventional methods.

2. Related Researches

The scheduling method of Wirth[5] is a centralized method. In this case, a central controller knows whole status and position of robot and assigns job to suitable robot. He categorized conventional methods to one-to-one, one-to-many, many-to-one, and many-to-many methods according to the number of departure and destination. Gunsser[4] suggested also a centralized method. But a controller assigns job to robot according to the sequence of its occurrence regardless of status of robots. Egbelu and Tanchoco[2] suggested heuristic rules which select one of robots which are idle. They divide task assignment rules to workcenter-initiated rules and vehicle-initiated rules. In Egbelu[3], he suggested a demand-driven dispatching rule. It is a method that workcenter to which object is transferred requests service instead of workcenter from which object is transferred requests service.

3. A Scheduling Method of Multiple Mobile Robot System

3.1. Problem formulation

We define some terminologies.

- Station: some position in workspace
- Job: transportation of a load from one station to another station
- Service Robot: robot to transfer load between stations

Consider an office environment shown in Fig.1. If one station has a load to transfer to another station, it requires service of robot. In this case, the job is to transfer a load from a station to another station. Therefore, we need the following information about each job.

- Request time: when station requests service
- Start station: from which load is transferred
- Target station: to which load is transferred
- Deadline: until when job has to be finished

If service request occurs, a robot moves to the start station and transfers the load from the start station to the target station. In this environment, service requests happen concurrently and irregularly, and independently from the stations.

We use two performance indices to evaluate the performance of multiple mobile robot system.

(1) Mean wait time
$$\sum (A_i - R_i)/n \qquad \text{for all job } i$$

(2) Mean tardiness
$$\sum \max(0, L_i)/N_t \qquad \text{for all job } i$$

where,

n : the number of jobs
A_i : time when robot arrives at the start station
R_i : time when station requests a service of robot
C_i : time when job is finished
D_i : deadline of job i
L_i : tardiness of job i $(= C_i - R_i - D_i)$
N_t : the number of jobs which is not finished until deadline

In this paper, we suggest a scheduling method that improves those performance indices.

3.2. Proposed scheduling method

The proposed scheduling method is an on-line and
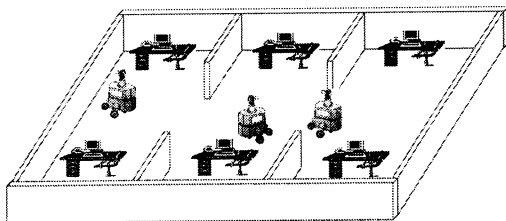
Fig. 1 The environment such as office

distributed method. Robots and stations communicate each other according to the format so called a coordination protocol. They negotiate and vote by it.

The coordination protocol is based on DEDS(Discrete Event Dynamic System) theory. We define important events and states. The events are the occurrences of new request, transmission of some specific messages, and so on. And, the state represents the status of robots or stations. If some event occurs, the state of a robot varies and the robot sends and receives messages to exchange state information with other robots. Then, the robot decides a task to perform next time

At first, we define the states of a robot. The robot is in one of three states: idle, move, and transport.

- Idle state: when the robot finished an old job and is waiting a new job
- Move state: when the robot is moving to the start station to take a load
- Transport state: when the robot is transporting a load to the target station

There are four events as listed in Table 1. The followings are time when each event occurs and what to do then.

A. "NewJob" event

"NewJob" event occurs when there is a new job in the station. In this case, the robots and stations take the following actions.

- Station (which has a new job)
1. It sends message to all robots. That message includes request time, start station, target station and deadline.
- Robot
1. If robot receives the message from the station, it stores the information of a new job to the job-list and determines the priorities of the jobs according to the task assignment rule. If the state of the robot is 'Idle' or 'Move', the robot returns to idle state. Otherwise, the robot continues current job.

B. "Idle" event

"Idle" event occurs when the state of robot is idle.

- Robot (the state of which is idle)
- If a robot has one more jobs in the job-list, it sends "Proposal message" to the start station of the highest priority job. That message includes a minimum execution time to take to finish it.

C. "Proposal" event

Table 1 Events

Event Name	When	Where
NewJob	A new job occurs	Station
Idle	A status of robot is idle	Robot
Proposal	Proposal comes from robot	Station
Answer	Answer comes from station	Robot
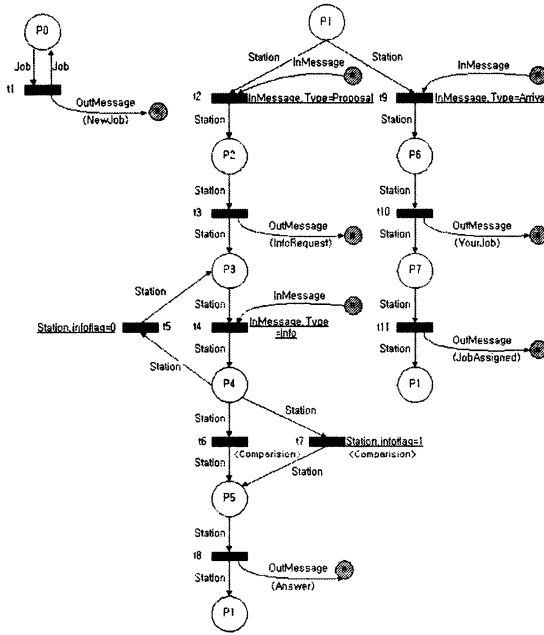
Fig. 2 Station model

"Proposal" event occurs when a station receives a proposal message from a robot.

- Station (which receives message)
1. It requires the minimum execution time to other robots except the robot that has sent proposal message.
2. If the station receives information from all robots, it determines a robot that has the smallest minimum execution time.
3. If the robot that has the smallest minimum execution time is robot that has sent proposal message, station sends "yes" to this robot. Otherwise, the station sends "no" to robots that have sent the proposal message.
4. Station returns initial state.

D. "Answer" event

"Answer" event occurs when station sends "Answer message" to the robot that has sent proposal message.

- Robot (which receives "Answer" message)
1. If "Answer" message is "yes", the robot moves to the start station to process this job. Otherwise, it sends "Proposal message" to the start station of the next higher priority job.

4. System Modeling

We need analyze the proposed method to check the resource conflict and deadlock. Multiple mobile robot system is DEDS like a network, a traffic system, etc.. We can model DEDS in a state machine, a queuing model, Markov chain, a Petri net, and so on. Petri net is popularly used as a model of DEDS because it suitably
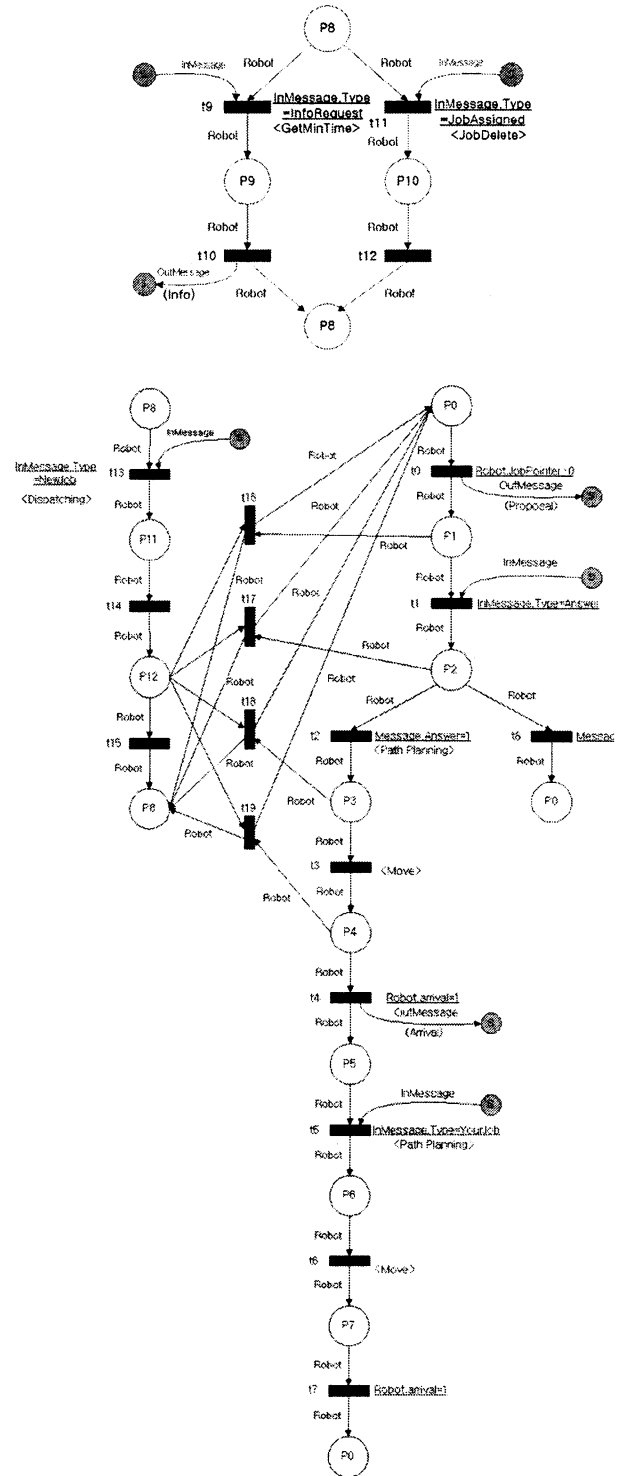
Fig. 3 Robot model

model the concurrency, synchronized event, resource conflict, logical sequence and structural relation of the system. Thus, we modeled the proposed method by Petri net.

We use Predicate/Transition net (Pr/Tr net) in many Petri net. Pr/Tr net increases modeling power by a transition condition and an action function. Figs.2 and 3 are the station model and the robot model. In these

models, gray color describes a communication port between robot and station. Each transition has the conditional equation and the action function. The conditional equation needs to be satisfied for the transition to be fired. When the transition is fired, the action function is activated.

5. Simulation

Fig.4 is a simulator that simulates the following environment.
- The size of workspace: 22.5m × 22.5m
- The size of station: 1.5m × 1.5m
- The radius of robot: 0.5m
- The velocity of robot: 15m/min.
- The load capacity of robot: 1 load
- The number of stations: 3
- The number of robots: 2~5
- The number of jobs: 15

Path planning of a robot is based on the cell decomposition method of Lim[1]. Robot plans a path and moves without collision.

We compared the proposed method with a conventional method as the number of robots increases from two to five. Task assignment rules of robots are FCFS(First Come First Serve) and EDF(Earliest Deadline First). In the conventional method, a central controller creates a schedule by FCFS or EDF. We carried out computer simulation 100 times each case.

5.1. FCFS task assignment rule

In FCFS rule, if a faster job happens, higher priority is assigned. It decreases the mean wait time. In this case, we use the mean wait time as a performance index. The result of the mean wait time is shown in Fig.5.

5.2. EDF task assignment rule

In EDF rule, if a job has a faster deadline, higher priority is assigned. It decreases the mean tardiness. Thus, we use the mean tardiness as a performance index. Result is shown in Fig.6.

6. Conclusion

In this paper, we considered the scheduling problem
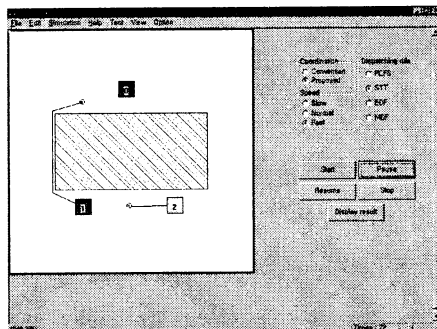
Fig. 4 Simulator
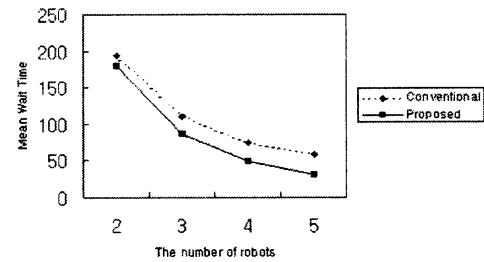
Fig. 5 Result for FCFS rule
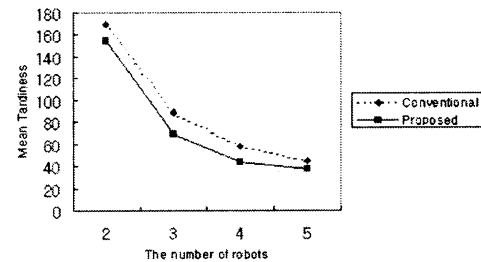
Fig. 6 Result for EDF rule

when robots perform jobs that happen concurrently and irregularly. In order to solve this problem, the scheduling method using the coordination protocol has been suggested. In this method, the robot initiated task assignment rule as well as the station initiated task assignment rule is used. Therefore, it decreases the mean wait time and mean tardiness. Since the proposed method is an on-line and distributed scheduling method, it is suitable for the dynamic environment. Furthermore, it shows an advantage of a fault-tolerance.

As a further work, more realistic simulation and experiments with real multiple mobile robot system are needed.

References

[1] J. K. Lim, "A study on static and dynamic path planning for autonomous vehicle", *IEEE International Workshop on Robot and Human Communication*, pp. 22-27, 1997.
[2] P. J. Egbelu and J. M. Tanchoco, "Characterization of automatic guided vehicle dispatching rules", *International Journal of Production Research*, Vol. 22, No. 3, pp. 359-374, 1984.
[3] P. J. Egbelu, "Pull versus push strategy for automated guided vehicle load movement in a batch manufacturing system", *Journal of Manufacturing System*, Vol. 6, No. 3, pp. 209-221, 1987.
[4] P. Gunsser, "Contol techniques in guided vehicle systems: Examples of applications and various control concepts", *Proceedings 2nd International Conference on AGVS*, pp. 85-113, 1983.
[5] P. Wirth, "Control system for robovehicle installations: Application-oriented in spite of standardization", *Proceedings 2nd International Conference on AGVS*, pp. 61-72, 1983.

An On-Line Learning Method for Object-Locating Robots using Genetic Programming on Evolvable Hardware

Ho-Sik Seok, Kwang-Ju Lee, Je-Gun Joung and Byoung-Tak Zhang
Artificial Intelligence Lab (SCAI)
Dept. of Computer Engineering
Seoul National University
Seoul 151-742, Korea
{hsseok, kjlee, jgjoung, btzhang}@scai.snu.ac.kr

Abstract

Evolvable hardware is a new concept of FPGA which has a capability of dynamic reconfiguration during run time. Due to its dynamic reconfiguration ability, evolvable hardware can optimize itself through learning. In this paper, we present a method for learning robot controller on evolvable hardware. For learning, we employ genetic programming. Typically, genetic programming uses tree-structured representation. However, tree structures are inconvenient for crossover in hardware and tend to consume much resource. Therefore, we use a linear chromosomes to represent genetic trees on evolvable hardware. The learning objective of the robot is to locate a light source avoiding obstacles.

1 Introduction

Evolvable hardware is a new concept of FPGA (Field Programmable Gate Array) which has a capability of dynamic reconfiguration during run time. Due to its dynamic reconfiguration ability, evolvable hardware is applicable to many areas such as fault-tolerant systems and adaptive systems [1]. Various evolutionary algorithms have been used as evolutionary mechanisms for evolvable hardware. Many researchers used binary-string genetic algorithms. Due to their advantage that the configuration bitstreams can be regarded as chromosomes. Alternatively, genetic programming can be used for evolving codes on evolvable hardware.

Since genetic programming usually uses tree structures and trees are flexible, genetic programming is appropriate for evolving complex-problem solving strategies [2]. We attempt to combine the advantages of evolvable hardware and genetic programming. Robot behavior can be described as a mapping from sensor inputs to motor outputs. Therefore, after determining

the inputs and outputs of a robot, genetic programs can be used to represent the mapping. Through re-ordering nodes of the genetic tree, the control structure of the robot can be adapted.

By combining genetic programming and evolvable hardware, we were able to construct a robot controller which can be adapted to environmental changes. In our approach, robot controller evolves its control structure using environmental data on evolvable hardware.

The paper is organized as follows. Section 2 reviews related work. Section 3 describes implementation details. In Section 4, some experimental results are shown. Section 5 summarizes the result and points out some future work.

2 Related work

There are many attempts to control an autonomous robot by genetic programming. Ebner evolved the control structure of a real mobile robot using genetic programming [3]. Wilson evolved hierarchical behaviors to locate a goal object in a maze for a mobile robot [4].

Some authors have attempted to evolve emergent collective behaviors using genetic programming. Bennett III used genetic programming to evolve a common program that controls foraging for food by ants [5]. Zhang introduced a framework, called fitness switching, that facilitates evolution of composite emergent behaviors of a multiagent system using genetic programming [6]. All these works have used software genetic programming as an evolutionary engine.

Genetic algorithms are frequently used as a mechanism for evolving hardware circuits. Here, configuration bitstreams of evolvable hardware are represented as chromosomes. Keymeulen used a genetic program-
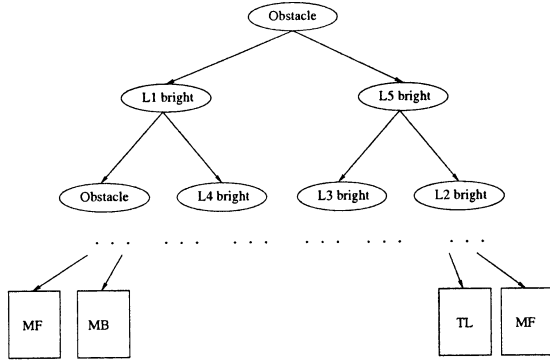
Figure 1: An example of a genetic tree. This tree represents a general control structure of an autonomous robot. The meaning of the left subtree is "if an obstacle is found, light sensor L1 is bright, and other conditions are met, then move forward (MF)."
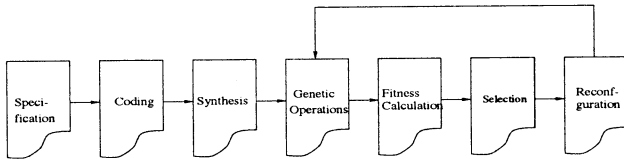
Figure 2: The procedure for hardware evolution. Starting from a population of initial circuits, the circuits are improved by repeating the evolutionary design steps.

ming algorithm for building a navigation system for an autonomous robot [7]. Thomson used genetic algorithms to design electronic circuits automatically [8].

Genetic programming has also been used for evolvable hardware. Sakanashi applied genetic programming to digital circuit design: Evolution of binary decision diagrams. He used genetic programming to improve the hardware description in binary decision diagrams [9].

3 Hardware evolution of genetic trees

3.1 Robot control using genetic programming

After specifying each sensor's input range and robot's output actions, the elements that determine the input-output of control programs are easily obtained. By transforming these elements to corresponding nodes of genetic tree, the preparation for genetic programming representation of control programs is completed.

Fig. 1 shows a control structure of a robot that finds a light-source while avoiding obstacles. This control program interprets environmental data as follows.

First, it determines if there is an obstacle. Then, it goes to one of the two subtrees. At the subtree below the root node, it determines which light sensor directs to the light source. By repeating above procedure, it reaches one of the terminal nodes. Then, the terminal node selects one of the possible actions of the robot. The objective of genetic programming is to find an optimal tree structure that controls the robot to the target position.

Fig. 2 shows the procedure for hardware evolution. First, the designer specifies the circuit that he wants. Second, the designer writes some initial code. Third, the design is synthesized by using a commercial CAD program. Then, the following evolutionary steps are repeated until a termination condition is satisfied: genetic operation, fitness calculation, and selection.

3.2 Hardware evolution of genetic trees

While there have been several attempts to implement bitstring genetic algorithms on evolvable hardware, relatively few attempts have been made to implement genetic trees on evolvable hardware. The usual GP tree structures can directly be represented on evolvable hardware as shown in Fig. 3. But, in this approach, the designer faces several problems that do not occur in software genetic programming. First, due to partial tree stucture, only restricted crossover is allowed. Second, a dominant part of hardware resources are consumed by routing. Third, a significant part of hardware resource is never used for placement nor for routing. To overcome these problems, we use a linear representation scheme.

We use linear strings to represent trees on evolvable hardware. To represent a genetic tree in binary strings, we separate each path from the root node to each terminal node, then each path is transformed to a binary string. Binary strings are of the same length. By using a linear representation, several advantages can be obtained. First, crossover operator can easily be implemented. Second, we can save hardware resources for routing. Third, we can increase the total resources utilization ratio. In our preliminary experiments, we were able to save the resources for routing by 12%, for example.

The terminal and function nodes we adopted are shown in Table 1. The terminal nodes denote robot actions. These are move forward (MF), move backward (MB), move forward and turn left (MTL) and move forward and turn right (MTR). The function nodes interpret sensor inputs. They consist of two different kinds of IF-statements:
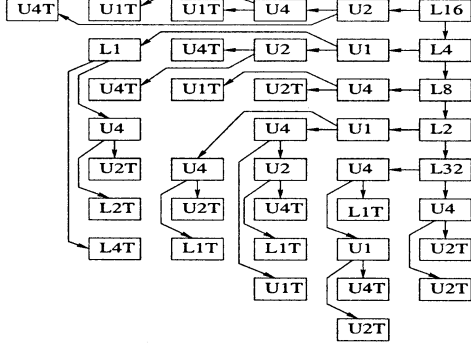
Figure 3: An example of hardware representation of a tree-structured chromosome. This representation scheme does not assure the same depth for all subtrees.

Table 1: Functions and terminals.

	Symbol
Terminal nodes	MF, MB, MTR, MTL
Function nodes	IF-L0, IF-L1, IF-L2
	IF-L3, IF-L4, IF-L5
	IF-US0, IF-US1, IF-US2

- IF-L0 ~ IF-L5: the intensity of the light sensors.

- IF-US0 ~ IF-US2: the ultra-sonic sensor values indicating the distance to an object.

4 Experiments and results

4.1 Evolution of robot controllers

The structure of the robot controller is determined through evolution of genetic trees. For efficient evolution, the evolution procedure is divided into two stages. In stage one, reordering of function nodes is performed. The function set consists of 9 function symbols. We use 4-bit binary string to represent each function nodes. Since we allow trees of depth 8, we should maintain 128 binary strings. In stage one, we repeat tree reordering until all interpretation paths of environmental data are obtained. Fig. 4 shows the fitness curves which indicate the advantage of using linear representation. By using linear representation, convergence speed is increased by a factor of approaximately three.

In stage two, proper behavior to each sensor-input is determined, and the evolution procedure is combined
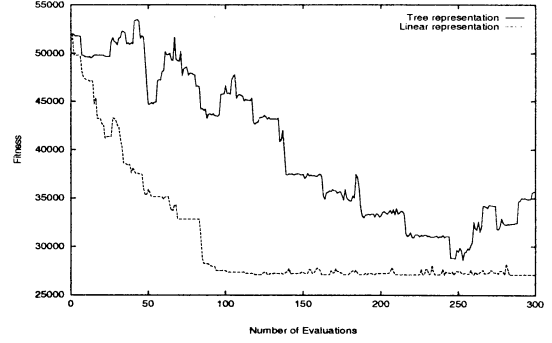
Figure 4: Comparison of experimental results for tree and linear representations. For tree representation, mutation was used as the only genetic operator since crossover caused difficulties in hardware implementation. For linear representation, both mutation and crossover were used.
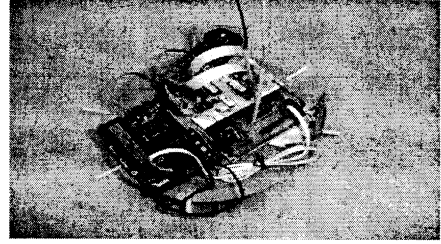
Figure 5: The robot used in the experiment.

with robot move. The result of movement by the previous control structure is used for reconfiguration of the genetic trees.

4.2 Experimental results

The robot has six light sensors for detecting the goal and three ultra-sonic sensors for estimating the distance between the robot and the obstacles. The learning objective of the robot is to locate a light source while avoiding obstacles. The robot is shown in Fig. 5.

Fitness of the robot controller was evaluated as follows:

$$
\begin{aligned}
Fit(t+1) &= [Fit(t) + (L_{max} - L_{fw})W_L \\
&+ (\frac{2U_{fw} + U_l + U_r}{2U_{max}})k \times W_U \quad (1) \\
&+ Penalty]/2
\end{aligned}
$$

Where the meaning of the symbols are $Fit(t)$: Fitness value at step t, L_{max}: Maximum input value of the light sensor, W_L, W_U: Weights of the light and ultra sonic sensors, k: Scaling factor, U_{max}: Maximum input value of ultra sonic sensor, U_{fw}, U_l, U_r: Estimated distance of the forward, left, and right ultra sonic sensor.
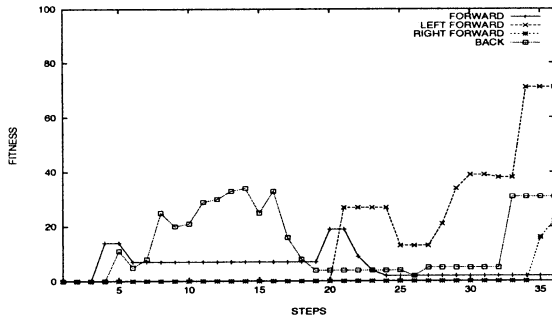
Figure 6: This graph shows the fitness curves about a sensor input when the left light sensor is the most intensified and the forward ultra sonic sensor estimated the longest distance to an obstacle. As known from fitness curves, the action MTL has the largest fitness value.

We assign more weights on the light sensor inputs since the ultra sonic sensor input may be affected by noise. By modifying W_L and W_U, we can change the objective of the robot. By assigning more weight on W_L, the learning objective of the robot is determined as locating the goal.

Fig. 6 shows the learning result of the robot. At 35th generation, the robot found the proper behavior (MTL) for the given sensor input pattern. As generation goes on, the robot finds proper behaviors for each sensor input pattern.

5 Conclusion

We use genetic programming for evolving a robot control structure. For implementation of genetic trees on evolvable hardware, we take a linear representation scheme. Linear representation has several advantages. First, it is easy to implement crossover operation. Second, it uses less resources for routing.

We applied our representation scheme to an object locating robot. Initially, a set of randomly structured genetic trees is created. As robot continues wandering, the robot uses sensor inputs to evolve proper control structures. For efficient evolution, we divided the evolution procedure into the interpretation path stage and the behavior stage.

Due to the limit in hardware resource, we evolved relatively simple control structure. For evolving more complex behaviors, future research on compact representation of genetic trees on evolvable hardware is required.

References

[1] L. Paul, "The 'evolvable motherboard': A test platform for the research of intrinsic hardware evolution," Cognitive Science Research Paper 479, 1998.

[2] J. R. Koza, "Genetic programming: On the programming of computers by natural selection," Cambridge, MA, USA, MIT Press.

[3] M. Ebner, "Evolution of control architecture for a mobile robot," International Conference on Evolvable Systems, pp. 303-310, 1998.

[4] M. S. Wilson et al., "Evolving hierachical robot behaviours," Robotics and Autonomous Systems, pp. 215-230, 1997.

[5] F. H. BennettIII, "Automatic creation of an efficient of an efficient multi-agent architecture using genetic programming with architecture-altering operations," Genetic Programming 1996: Proceedings of the First Annual Conference, pp. 30-38, 1996.

[6] B-T. Zhang and D-Y Cho, "Fitness switching: Evolving complex group behaviors using genetic programming," Genetic Programming: Proceedings of the Third Annual Conference, pp. 431-438 1998.

[7] D. Keymeulen, "An evolutionary robot navigation system using gate-level evolvable hardware," International Conference on Evolvable Systems, pp. 195-209, 1996.

[8] A. Thompson, "An evolved circuit, intrinsic in silicon, entwined with physics," International Conference on Evolvable Systems, pp. 390-405, 1996.

[9] H. Sakanashi et al., "Evolution of binary decision diagrams for digital circuit design using genetic programming," International Conference on Evolvable Systems, pp. 470-481, 1996.

Generation of Rules for Swarm Intelligence of Autonomous Mobile Robots

Seo-Kwang Kim and Seong-Gon Kong

Intelligent Signal Processing Lab.
Department of Electrical Engineering
Soongsil University
Seoul, 156-743 Korea
E-mail : dolmangi@ee.ssu.ac.kr, skong@ee.ssu.ac.kr

Abstract

Complex lifelike behavior is considered as local interactions between simple individuals under small number of fundamental rules of multiple autonomous mobile robots. This paper presents a novel approach for coordinating a homogeneous system of mobile robots using implicit communication in the form of broadcasts. The broadcast-based coordination scheme was developed for the Army ant swarm. Coordination of the swarm is achieved with signals we call heartbeats. Each agent broadcasts unique "heartbeat" and responds to the collective behavior of all other heartbeats. In this paper we introduce the protocol of "heartbeat" and the algorithm of swarm intelligence by "heartbeat" protocol.

1. Introduction

Autonomous mobile robot makes a decision by itself and moves freely to any direction based on sensor inputs.[1] A few autonomous mobile robot which have simple capacity with swarm intelligence solve the complex problem that is not settled by a individual.[2] The swarm intelligence emerges some complicated functions which are made by a few individuals with simple ability in cooperative behavior.[3][4] The group of autonomous mobile robots with swarm intelligence has some basic characteristics that are robustness based on homogeneity, independent invariant complexity to the number of robots, cheapness and compactness.[3]

To realize swarm intelligence, we use the fuzzy rules that represent a few fundamental rules of artificial life to govern group behavior of autonomous mobile robot.[4] The rules generated by input and output data. we merge fuzzy inference system into programmed simulator to adapt real robots and endow duty of material handling and evaluate the performance about material handling.

Keywords : Artificial life, Swarm intelligence, Fuzzy rule, Group behavior, Mobile robot

2. Fuzzy System for Swarm Intelligence

Generally, there are two cases to obtain the rules of fundamental artificial life. If expert knowledge exists, fundamental rules artificial life can be directly represented by linguistic fuzzy rules.[4][5] However, if fundamental rules numerical data are available expert-knowledge, fundamental rules that explain A-Life phenomenon can be acquired by fuzzy system modeling with clustering.

It is important to decide the number of fuzzy rules and rules in fuzzy modeling and those are obtained by clustering. Clustering is the algorithm which is able to classify some group depend on similarity in data. The number of clusters are the number of fuzzy rules and cluster centers determine the type of fuzzy rules.[8]

3. Material Handling

3.1. Group behavior for material handling

Material handling is to handle some materials scattered in open space into destination by multiple autonomous mobile robots. Four states are necessary to handle the material because of a series of dealing procedure and each state can operate by using module of fuzzy system.
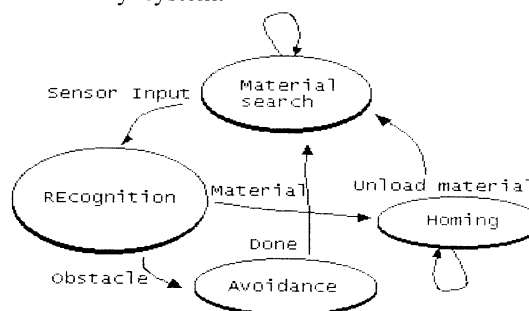
Fig 1: State diagram of material handling

On performing the duty, robot goes toward at the same time searches the material. If an object is appeared, robot judges whether it to be obstacle or not. If robot decides the object to be obstacle, the robot adapts evading obstacle algorithm and If robot decides the object to be material, the robot raise the material using the gripper and go back HOME. when the robot meets with some objects during going back

home, unconditionally the robot judges those to be obstacles and quickly returns HOME.

A robot which finds material at first informs the other robots in neighbor locations to the material by means of Broadcast way. Some robots that received the signal move to find the material face to the signal and the others find other material peculiarly without any information. Each robot has no information about coordinates so, it can infer self-position, find the goal and go back to bring the residual material using 8 sensor inputs.

3.2. Communication for Group behavior

Autonomous mobile robot is an independent agent that can move freely as its own judgement but each robot's function is very restricted and there's no explicit method communicating with other robots. but it can use implicit communication method like a broadcast. The Broadcast is the optimal communication method that fits to objective of the artificial life,[6] that is, the emergence of complex cooperation by mutual operation of simple agents.

The Broadcast is a one-sided communication method that one sends signal for others and there's no data exchange. So this is not an explicit and organized communication method but this is like the group behavior of real insects like ants.

4. Reference Algorithm and Fuzzy System for Swarm Intelligence

4.1. Reference algorithm for Recognition

When the robot meets a obstacle, it needs to identify materials the wall or robots. A robot can identify a material from the different values of 4 sensors in front of the robot as in Fig 2.

In Fig 2(a), when a robot is in front of wall, the difference values of each sensors are small, (b) when a robot meets other robot, the difference value of each sensors are bigger than (a). And if a robot encounters a small material, there's only one sensor input or same amount of sensor value between adjacent sensors.
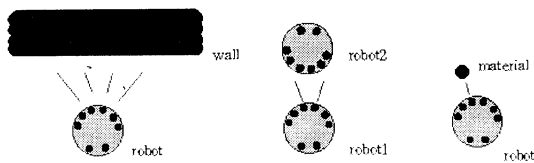
(a) wall　　(b) robot　　(c) material
Fig 2: Recognition of the Material

4.2. Fuzzy system for Recognition

To design a fuzzy system model for recognition of materials, in this study, first I have measured sensor input data in front of each materials according to distance for 10 times. And clustered the data to get 19 fuzzy membership functions for each input and 4 fuzzy output, that is, unknown(0), wall(1), wall(2), material(3).

(a) Sensor1 (s1)　　　　(b) Sensor 2(s2)

(c) Sensor3 (s3)　　　　(d) Sensor4 (s4)
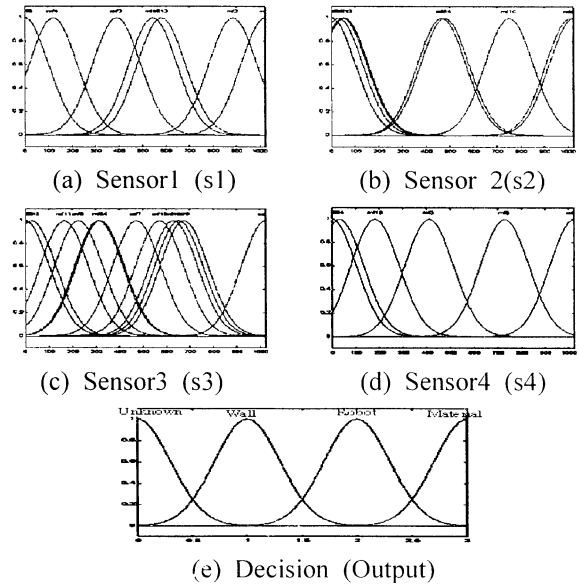
(e) Decision (Output)
Fig 3: Membership function for recognition

4.3. Reference algorithm for Avoidance

The algorithm of obstacle avoidance uses each sensor input values with different weights and that determines the motor's rotating direction and speed like the formula (1) and (2).
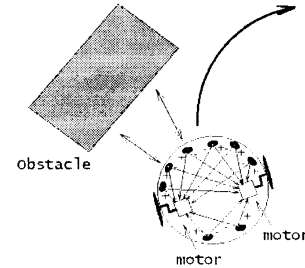
Fig 4: Avoidance based on sensor-weight

$$v_L = u^T s + v_F \tag{1}$$

$$v_R = w^T s + v_F \tag{2}$$

$$s = [s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7] \tag{3}$$

$$u = [4, 4, 6, -18, -15, -5, 5, 3] \tag{4}$$

$$w = [-5, -15, -18, 6, 4, 4, 3, 5] \tag{5}$$

In formulas above, s is a vector of sensor input, u, w is the weight vector that determines left and right wheel's speed each. And v_F is a forward revolution bias that is a fixed speed when there's no sensor input.

4.4. Fuzzy system for Avoidance

To design a fuzzy system model for avoidance, I have made a robot travel an area with 16 obstacles and got 500 data of sensor and motor speed each and clustered then. As a result, I have found 15 cluster centers and made a MIMO fuzzy model that determines each motor's speed from 8 sensors. Table 1 is about the center values of fuzzy membership functions. The universe of discourse of input is 1 ~

1024 and 1024 means an obstacle is very close to robot. The Fuzzy output is each motor's speed and its range is -20~20. All input, output membership functions are of gaussian form and the width of all gaussian membership functions are set to 0.15.

Table 1:Center of membership Function

	Input								Output	
	s0	s1	s2	s3	s4	s5	s6	s7	L	R
mf1	8.0	6	7	6	7	8	9	24	0	0
mf2	8.0	6	7	6	7	8	9	24	0	0
mf3	187	44	6	3	6	5	2	8	1	-2
mf4	187	44	6	3	6	5	2	8	1	0
mf5	4	6	7	7	489	286	4	1	-3	-2
mf6	8	6	7	6	7	8	9	24	0	1
mf7	6	3	5	0	9	1	939	968	8	9
mf8	7	8	9	8	9	7	9	921	3	6
mf9	81	944	922	967	0	8	6	4	-9	-24
mf10	0	5	107	946	967	427	5	2	-42	13
mf11	5	7	506	940	263	4	7	5	-23	-32
mf12	6	220	865	2	4	2	3	8	7	-2
mf13	4	9	5	2	5	2	941	3	6	3
mf14	9	9	2	462	701	108	4	9	-24	7
mf15	187	44	6	3	6	5	2	8	1	0

5. Autonomous Mobile Robot Modeling

5.1. Kinematics of a 2-wheel robot

The mechanical model of bi-wheel mobile robot is presented in Fig 5. The speed to the surface is presented in formula (6) with the condition of non-slipping and pure rolling. ω_L and ω_R is the angular speed of left and right wheel.
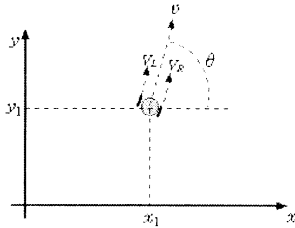
Fig 5: Coordinate of mobile robot

$$V_L = r\omega_L, \qquad V_R = r\omega_R \qquad (6)$$

Where each of parameters V_L and V_R is a motor's speed and r is a radius. (7) is a angular and a line velocity of a robot from the origin and (8) is a position change of a robot.

$$\omega = \frac{V_R - V_L}{L}, \qquad v = \frac{V_R + V_L}{2} \qquad (7)$$

$$\begin{pmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta}_c \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} \qquad (8)$$

Where - v is the liner velocity of center of the robot , w is the angular velocity of the robot to

counterclockwise , $\dot{x}_c$ means the change value of the X-Axis, $\dot{y}_c$ means the change value of the Y-Axis. And $\dot{\theta}_c$ means the change value of a robots direction angle. L represent distance between two wheels.

5.2. Modeling of Sensors

According as the distance, the measurements of Sensors dont increase linearly but as follows it did modeling as a linear approximation.
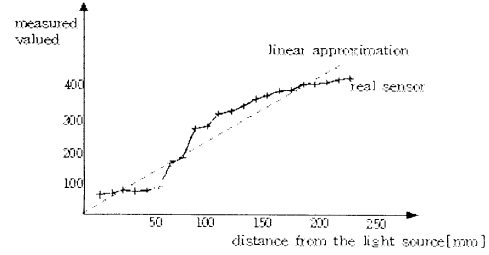
Fig 6: Approximation of the sensor

Eight sensors which are located in front and in rear from the center of a robot are attached to a robot. These have the available range of 10 Cm and the angle of the light of sensors is about $30°$. The range of measurement of an applicable sensor check the existence and nonexistence of materials

$$S_i = 10 \sum_{j=1}^{j=10}(10 - O_{j\text{distance}}) + r \qquad (9)$$

i : # No. of Sensor , j : No. of Scanline

$j$distance : Distance of the jth scanline

r : random noise.

5.3. Communication and Protocol

The medium of communication must be a wireless and considerable methods are RF, supersonic waves, infrared rays etc. But RF isn't strongly influenced by obstacles, other robots and the sunlight etc. Because it has no a sense of direction and cannot find the place being signalized. So it is excepted from a considerable object. Because infrared rays have advantages in the view of volume , consumption of power and less an interference phenomenon. Table 2 is a protocol used to communication.

Table 2: Protocol "Heartbeat"

0x5F	0xFA	1byte	1byte
Start byte	Confirm byte	Sender ID	Sender Status

The Sender ID is unique 1byte integer as fingerprint or face feature of a human. Sender Status is for inform other robot of its current status and Its number means that as Table 3.

Table 3: Status of Robot

No.	Status
0	Idle
1	Searching for the materials
2	Found the material
3	. Carrying the material
4	Searching for HOME
other	Reserved
255	Out of order

Four heartbeat sensors are located on the front, rear, left and right of the autonomous mobile robot to receive signal from other robot. A generated signal is spread toward eight directions around robots at a speed of 115200bps. After a robot lifts a material, it rotates right and left about 45 for spreading a uniform signal over whole sections and returns to the former place.

6. Simulation Results

6.1. Material Handling Simulation

For material handling simulation, 10 autonomous mobile robots are made to handle 100 materials in square space whose width is 2[m]. Fig. 7 shows material handling simulation with 10 robots and 100 materials in
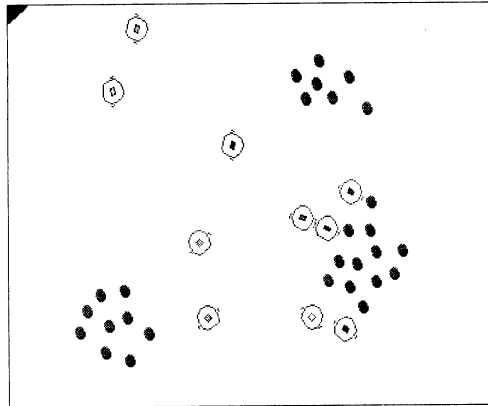
Fig 7: Material Handling Simulation

In order to evaluate the performance of swarm intelligence, we measured elapse time to handle 100 materials without communication for 3 times for increasing number of robots from 1 to 10. Fig. 8 shows the elapse time for handling materials is nonlinear and is not proportional to the number of robots.
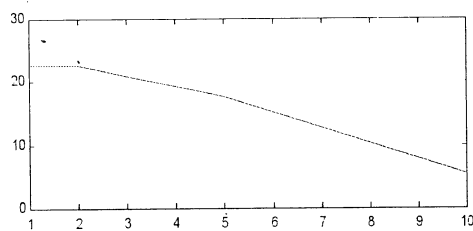
Fig 8: Elapse time vs number of robots

Table 4: Elapse time for different number of robots

# of robots experiment	1	2	3	5	10
1	22.34	22.01	20.33	17.10	5.28
2	20.21	19.45	20.41	18.54	4.00
3	25.40	26.22	22.27	17.35	7.08
Average	22.65	22.56	21.003	17.663	5.453

7. Conclusions

The fundamental rules of artificial life are represented by the fuzzy rules, generated from clustering input-output data obtained from the movement by the reference algorithm. The fuzzy rules are developed from numerical input-output data generated by the ideal reference algorithm of group behavior.

In this paper, the newly suggested heartbeat protocol is proven to be appropriate for swarm intelligence. Extensive material handling simulation using autonomous mobile robots show its advantages.

References

[1] J. Xiao, Z. Michalewicz, L. Zhang, and K. Trojanows ki, "Adaptive Evolutionary Planner/ Navigator for Mob ile Robots," IEEE Trans. on Evolutionary Computatio n, Vol. 1, No. 1, pp. 18-28, April 1997.

[2] M. Sipper, "An Introduction to Artificial Life," Explor ations in Artificial Life, pp. 4-8, 1995.

[3] J.S. Bay, "Design of the 'army-ant' cooperative lifting robot," IEEE Robot. Automat. Mag., vol 2., no.1m M ar.1995

[4] M.S. Kim, S.G. Kong, "Fuzzy Control System Design based on Data Clustering in Input-Output Space," Jour nal of the IEEK, vol.34, No.12, pp. 1370-1380, 1997.

[5] S.G. Kong, "A-Life and Fuzzy System," Journal of th e IEEK, vol. 24, No.3, pp. 319-327, 1997.

[6] K. S. Evans, "A Reactive Coordination Scheme for a many-robot system," IEEE Trans. On Systems, Man a nd Cybernetics, vol. 27 598-610

[7] M. Sugeno and G. T. Kang, "Fuzzy Identification of Systems and its Application to Modeling and Contro l," IEEE Trans. on Systems, Man and Cybernetics, V ol. 15, pp. 116-132, 1985.

[8] J. S. R. Jang, "ANFIS: Adaptive-Network- based Fuzz y Inference Systems," IEEE Trans. on Systems, Man and Cybernetics, Vol. 23, 665-685, May 1993.

[9] J.-H. Kim and S.-G. Kong, "Implementation of Group Intelligence Algorithm of Autonomous Mobile Robots by Clustering (Korean)," Journal of the KIEE, Vol. 4 7, No. 9, pp. 1532-1537, Sept. 1998.

Evolvable Cellular Neural Networks:
A New Paradigm for Classifiers

Hoon Kang
School of Electrical & Electronics Engineering, Chung-Ang University
221 Huksuk-dong, Dongjak-gu, Seoul 156-756, KOREA
Email: hkang@cau.ac.kr

Abstract

A cellular automaton is well-known for self-organizing and dynamic behaviors in the field of artificial life. This paper addresses a new neuronic architecture called an *evolvable cellular neural network* which evolves with the genetic rules (chromosomes) in the non-uniform cellular automata. An evolvable cellular neural network is primarily based on *cellular programming*, but its mechanism is simpler because it utilizes only mutations for the main genetic operators and resembles the Hopfield network. Therefore, the desirable bit-patterns could be obtained through evolutionary processes for just one individual agent. As a result, an evolvable hardware is derived which is applicable to classification of bit-string information.

Keywords: Cellular Automata, Artificial Life, Evolving Neural Networks, Classifiers, Evolvable Hardware

I. Introduction

Cellular automata are a class of discrete spatio-temporal devices in which the spatial states and/or the genes of each cell evolve with time. The behaviors of cellular automata are so complex that even chaotic dynamics could occur, and if conditions are satisfied, emergent behaviors might be shown[1]. Historically, one can find the origin of cellular automata from von Neumann's self-organizing machines[2]. Conway's famous game of life[3] is also a by-product of cellular automata which reveals the fact that local interactions may end up with global life-like behaviors. Codd's cellular machines[4] influenced deGaris' evolving cellular brain machines[5]. Wolfram[6] analyzed cellular automata and classified those into 4 categories, and Langton's loop[7] is another example that shows self-organization. Mitchell et al.[8] have investigated the dynamic behaviors of evolving cellular automata. Sipper[9] has solved several problems such as density, synchronization, and ordering by using his own co-evolved parallel cellular machines in terms of *cellular programming*.

This paper is concerned with cellular programming but is rather modified based on Hopfield networks[10] from the viewpoint of cellular automata. The main objective is to propose a new paradigm of evolvable neuronic architecture based on cellular automata which classifies binary information and solves genetic rules giv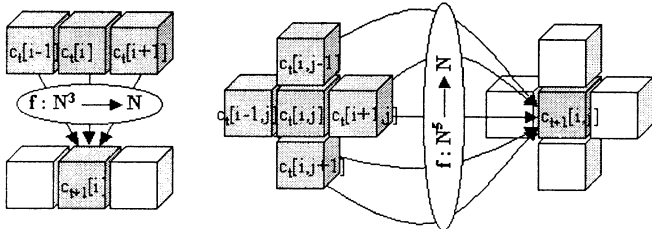en by the prescribed bit-patterns of the cell states. Therefore, given any distributions of initial states, the genetic rules that have the cell states converge to the exact bit-string, are evolved through mutations in non-uniform genetic rules with an evolvable cellular neural network.

(a) 1-D CA (3 neighborhood)   (b) 2-D CA (5 neighborhood)
Figure 1. Local Functions in Cellular Automata

II. Synthesis of Evolvable Cellular Neural Networks

◆**Non-uniform Cellular Automata**

First, consider the design procedure of an evolvable cellular neural network. Each cell state consists of 1-bit data in $\{0,1\}$ and usually, the number of the neighborhood is 3 in 1-D case and 5 (center, north, east, west, south) in 2-D case of the von Neumann neighborhood. The dynamical behavior of each cell is governed by an implicit rule set called a "gene", 8-bits in 1-D case and 32-bits in 2-D case, respectively, distributed in a non-uniform lattice. Figure 1 shows how a cell state is computed logically by using the neighborhood surrounding itself. The boundaries of cellular automata are cyclic in 1-D case and toroidal in 2-D case.

◆**Structure of Evolving Cellular Neural Networks**

The cell states in evolvable cellular neural networks are represented by

$$c_{t+1}[i] = f_{1D}(c_t[i-r], \cdots, c_t[i], \cdots, c_t[i+r]) \tag{1a}$$

$$c_{t+1}[i,j] = f_{2D}\begin{pmatrix} & c_t[i,j-1] & \\ c_t[i-1,j] & c_t[i,j] & c_t[i+1,j] \\ & c_t[i,j+1] & \end{pmatrix} \tag{1b}$$

where $c_t[i]$ is the i-th cell state at time t, and f_{1D}, f_{2D} are functions of 1-D cellular automata with $(2r+1)$ neighborhood, 2-D cellular automata with von Neumann neighborhood, respectively. The functional operations are performed in parallel at each time t. Each cell in cellular automata contains the genotypes of an 8-bit binary genetic rule in 1-D case, and a 32-bit binary gene in 2-D case, which are described by

$$\text{gene}_{t+1}[i] = g_{1D}(\text{local\_fitness}_t[i], \text{gene}_t[i], \text{other\_genes}) \tag{2a}$$

$$\text{gene}_{t+1}[i,j] = g_{2D}(\text{local\_fitness}_t[i,j], \text{gene}_t[i,j], \text{other\_genes}) \tag{2b}$$

where g is the genetic operator, gene_t is a genetic rule at time t, and local_fitness_t is a 1-bit local fitness value in $\{0, 1\}$ at time t. An optional input argument <other genes> is taken into account when co-evolved evoltionary processes are involved. The local fitness is obtained by the exclusive NOR (XNOR) operation given by

$$\text{local\_fitness}_t[.] = \text{XNOR}(c_t[.], \text{desired\_bit}[.]) \tag{3}$$

where desired_bit[.] is an arbitrarily chosen desirable bit-pattern in $\{0, 1\}$ related to the real-world phenotypical representations.
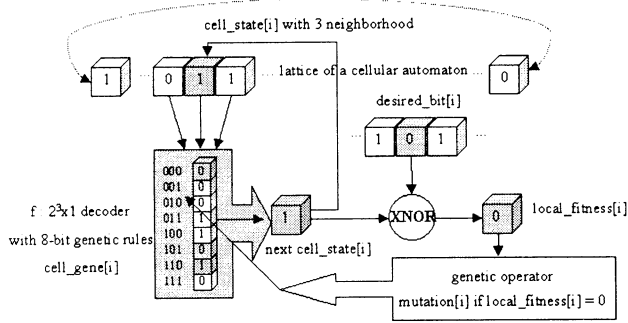
Figure 2. Block Diagram of Evolvable Cellular Neural Networks (1-D)

◆Evolutionary Processes in Evolvable Cellular Neural Networks

In brief, the block diagram of an evolvable cellular neural network is shown in Figure 2 in which the dynamic behaviors of cell states are controlled by neighboring cell states and non-uniform genetic rules, distributed randomly at the initial stage. The main genetic operator is a bit-wise partial mutation or inversion for each genetic rule. The evolutionary processes are divided into two phases, the evolving phase (A) and the test phase (B). (A) In the evolving phase, as the desirable bits are determined, an evolvable cellular neural network searches suitable genes that induce the desirable bit-patterns of the cell states. Even if the search is partially completed, all the local fitness values may be satisfied. (B) Therefore, it is required to continue to test an evolvable neural network by fixing the genes and perform parallel operations of non-uniform cellular automata. If at least one bit of the local fitness is not set during the test phase, it switches to the evolving phase in order to re-search the fittest genes. Since both processes are performed in the steady-state, the genetic changes in the transient response should be skipped by free-running for M time steps which is closely related to the size of the neighborhood. As the evolutionary processes continue, from any initial distributions of the cell states, the final cell states converge to the desired bit-patterns, and thereafter, only the genes of an evolvable cellular neural network may contribute to restoring the information.

◆Programming Evolvable Cellular Neural Network

The following pseudo-coded algorithm in Table 1 shows the detailed procedure of programming an evolvable cellular neural network, which is similar to Sipper's *cellular programming*[9], but modified appropriately to resemble the discrete Hopfield-type memories. Because the learning algorithm is replaced with the evolutionary processes, the initial random distributions of the cell states could be classified into some desirable bit-patterns. Also, note that it is possible to switch to the evolving phase at any time during the test phase. In the initial phase, the transient behaviors of the cell states should be preserved by fixing the genetic rules for some period of time. The proper value of the transient period M depends on the number of neighborhood since a cell state changes by referring to those cell states in the vicinity of itself. Intuitively, we could make an assumption that the steady-state response may be influenced in a triangular form of propagation in the nearby states. Therefore, $M > 3 \times (2r+1)$ (1-D case) or $M > 3 \times$(the number of the neighborhood) is a good choice of preserving the transient effects at the initial stage.

```
Evolvable Cellular Neural Network Programming:
set all desired_bit[i];
set random_genes;
random_configuration = 0;
while not(converged),
  set random_states;
  time = 0;
  free-run CA M-time steps with fixed non-uniform genes;
  evaluate all local_fitness;
  if (all local_fitness = 1), phase = 'test_phase';
  else phase = 'evolving_phase';
  end if;
  while not(all local_fitness[i] = 1),
    for all cell[i] do in parallel;
      update cell_state[i];
      if (desired_bit[i] = cell[i]), local_fitness[i] = 1;
      else local_fitness[i] = 0;
        if (phase = 'evolving_phase'), mutation(gene[i]);
        end if;
      end if;
    end parallel;
    if (at_least one local_fitness=0), phase='evolving_phase';
    end if;
    time = time + 1;
  end while;
  random_configuration=random_configuration+1;
end while;
```

Table 1. Programming Evolvable Cellular Neural Networks

◆Local Fitness vs. Global Fitness

Here, an interesting concept of the relationship between a local fitness and the global fitness arises, and it may establish the link from the encoded genotypes to the phenotype of feedback in the real-world environment. In our case, a simple local fitness is defined in eq.(3) which results from an error of local interaction between a cell state and the desired state, and the global fitness, in general, can be defined in eq.(4) as follows:

$$\text{global\_fitness}_t = h(\text{local\_fitness}_t[\cdot]) = \sum_{\forall i} a_i \times \text{local\_fitness}_t[i] \quad (4)$$

where h is an evaluating function, and a_i is a linear coefficient so that the global fitness may be the weighted sum of all the local fitness values.

◆Redundancy and Evolvable Neural Networks as Classifiers

The local distributions of the genetic rules are redundant in a sense that the same convergent bit-patterns are likely to have many different genetic codes. The reason results from the fact that the dynamic behaviors of the same convergent bit-patterns of the cell states change with a wide variety of sets of genetic rules which eventually perform as a many-to-one mapping. Therefore, we are able to store a collection of desirable bit information under different initial conditions in the nearby Hamming distance. It means that evolvable cellular neural networks are applicable to classifiers which appropriately separate the Hamming space, in which similar coding problems of the existing neural networks in [11,12] are dealt with as in associative memories. Only difference is the fact that the learning process is supervisory and evolutionary. In Table 1, the algorithm is slightly modified to include several desired bit-patterns at the same time as the evolution proceeds. That is, for each configuration, the multiple initial distributions corrupted by

bit-reversal noise are applied simultaneously to the associated states with the same genetic rules. The local fitness evaluation is also changed in such a way that a local fitness value in each cell is set to 1 only if every state values of the cell at the same location for different desired values are satisfied. This is due to redundancies in producing 0's and 1's of each gene as mentioned above.

III. Main Results : Online Evolution of Evolvable Cellular Neural Networks

The initial densities of the cell states are chosen so that the probability of 1's is 0.5 (p=0.5) which may vary in some applications. For example, in a pattern classfication problem, the desired bits can be disturbed by random noise and these may be prepared for the initial distributions. The maximally allowed mutation probability is 100% and the mutation rate per gene is varied from 1/8 to 1/2 in both 1-D or 2-D cases. Also, the free-running period M is 15, and a_i's in eq.(4) are chosen 1/L where L is the lattice size (L_{1D}=64, L_{2D}=16x16 or 32x32).

◆ **Examples of 1-D Evolvable Cellular Neural Networks**

Two desirable bit-patterns, 'Yin-Yang' and 'Stripes', are applied to 1-D evolvable cellular neural network. The evolving phase of the initial (0th) configuration is shown in Figure 3-(a) for the 'Yin-Yang' pattern where the cell states (left-top), the evolving genetic rules (right-top), the local fitness values (left-bottom), and the global fitness (right-bottom) are shown. Figure 3-(b) represents the same graphs as in Figure 3-(a), but the evolving phase of the 1st configuration is shown where the frequency of mutations is reduced conspicuously. Note that the number of changes in genetic rules is decreased both in time and in size. In Figure 3-(c), it is shown that the evolving and test phases take turns in order for the genetic information to converge under the random configuration no.'s 2, 3, 5, 25, 34, & 68. Figure 4-(a) also demonstrates the evolving phase of the initial (0th) configuration for the 'Stripes' pattern. Similarly, Figure 4-(b) and Figure 4-(c) show the evolving phase of the 1st configuration and the combined phases under the random configuration no.'s 2, 3, 5, 25, & 77, respectively. For the testing purpose only, the cell states and the associated genes are represented in Figure 5, in which the fixed genes are applied as the results of runs for each random configuration. Figure 5-(a) and Figure 5-(b) show the results of the 'Yin-Yang' test pattern and the 'Stripes' pattern, respectively. As the mutation rates per gene change, the average global fitness values of the test pattern no.2 ('Stripes') are compared in Figure 6 where the mutation rates per gene are varied 1/8, 2/8, 3/8, and 4/8. Although it is not obvious to see that a large mutation rate per gene results in better performance, the final trend reveals that the global fitness converges to 1 faster if the mutation rate per gene is larger.

For a long-term evaluation of evolvable cellular neural networks, we may consider the information aspect of the entropy concept[13]. Let p be the probability of success (say, '1') in a cell state and let T be the observation time, then the entropy $E(p)$ of the binomial distribution becomes

$$E[p(T)] = -p(T)\log_2(p(T)) - (1 - p(T))\log_2(1 - p(T)) \quad (5)$$

where the probability $p(T)$ in each cell is calculated during the observation time T. Here, in Figure 7, the spatio-temporal entropy values are compared between two test phases in the 0th and the 80th configurations of a 1-D evolvable cellular neural network. The result of the 80th configuration shows less entropy

values, which means that the randomness in the cell structure reduces as the evolutionary phase proceeds, in accordance with our anticipation.
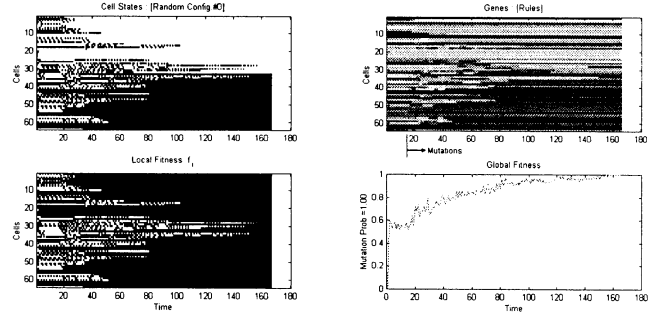
Figure 3-(a). Evolutionary Process of the 0th Config. in 1-D (Yin-Yang)
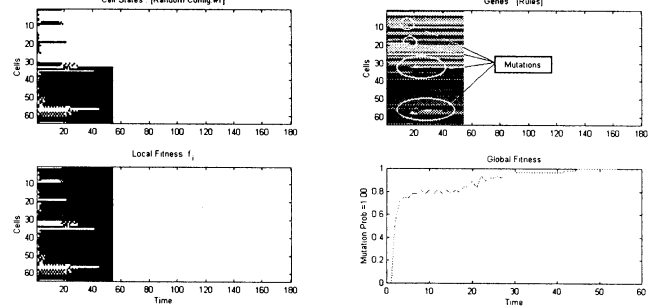
Figure 3-(b). Evolutionary Process of the 1st Config. in 1-D (Yin-Yang)
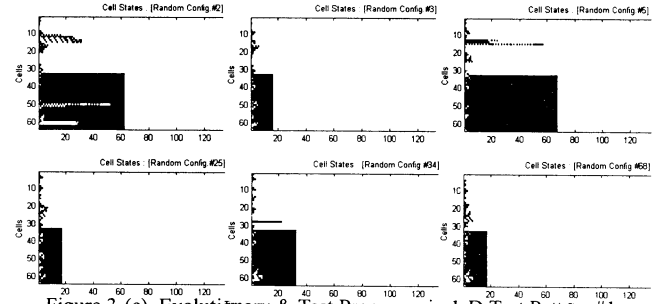
Figure 3-(c). Evolutionary & Test Processes in 1-D Test Pattern #1 (Yin-Yang) with Random Config. No.'s 2, 3, 5, 25, 34, & 68

In Figure 8, an evolvable cellular neural network plays a role as a pattern classifier to which two complementary test patterns of 'Yin-Yang' are applied where two initial conditions converge to the two desirable complementary 'Yin-Yang' patterns. It demonstrates how one set of genetic rules may produce different cell states from the various initial distributions located in the nearest Hamming distance. Here, the initial states are disturbed by 10% bit-reversal noise each. As mentioned earlier, redundancy in the cell chromosomes is the primary reason for the classification mechanism. Therefore, an evolvable cellular neural network can be used for classification problems.
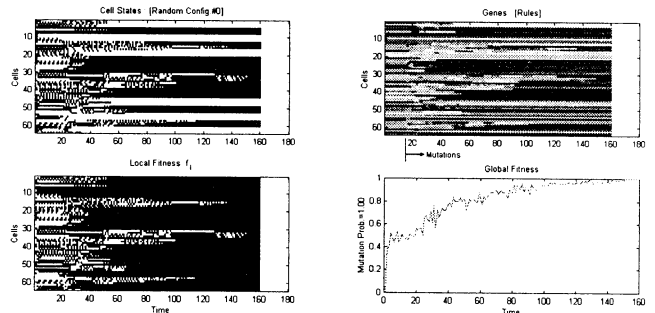
Figure 4-(a). Evolutionary Process of the 0th Config. in 1-D (Stripes)
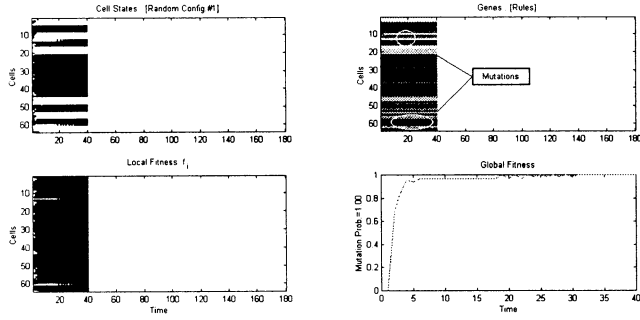
Figure 4-(b). Evolutionary Process of the 1st Config. in 1-D (Stripes)
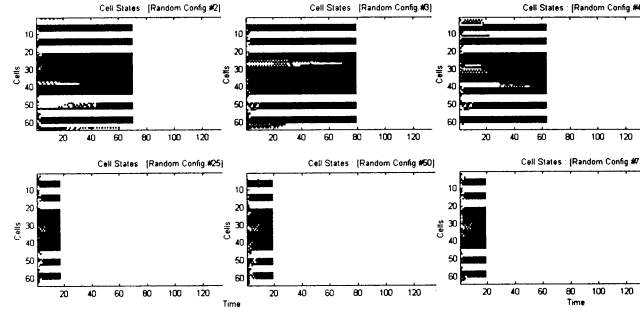
Figure 4-(c). Evolutionary & Test Processes in 1-D Test Pattern #2 (Stripes) with Random Configuration No.'s 2, 3, 4, 25, 50, & 77
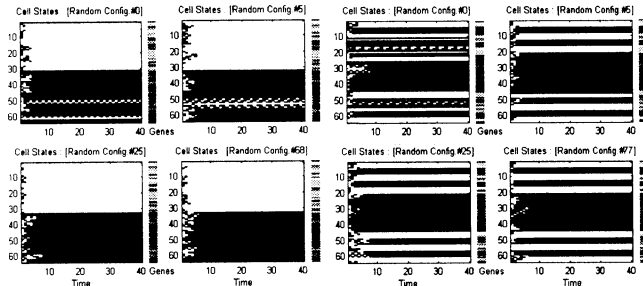
(a) Configuration No.'s 0, 5, 25, 68 (b) Configuration No.'s 0, 5, 25, 77

Figure 5. Test Processes of Config.'s in (a) Yin-Yang (b) Stripes
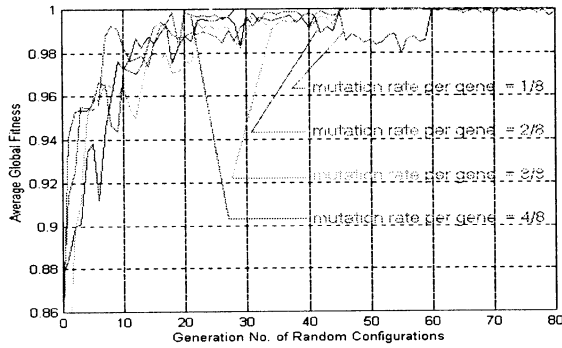
Figure 6. Comparison of Average Global Fitnesses in Stripes
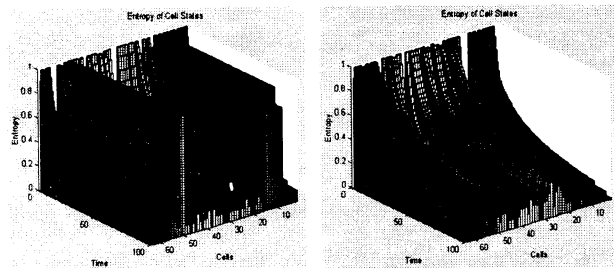
(a) Cell Entropy (0th Config.)   (b) Cell Entropy (80th Config.)

Figure 7. Spatio-Temporal Entropy in 1-D case
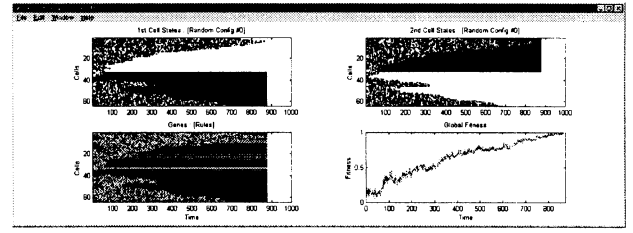
(a) Evolutionary Classification Process of the 0th Configuration in 1-D Two Test Patterns (2 Yin-Yang's)
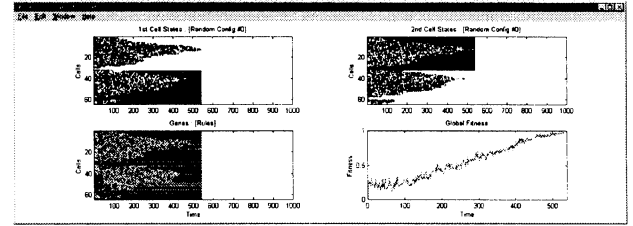
(b) Evolutionary Classification Process of the 0th Configuration in 1-D Two Test Patterns (2 Yin-Yang's)

Figure 8. 2 Cases in Evolving Phase as a Pattern Classifier

◆ **Examples of 2-D Evolvable Cellular Neural Networks**

In 2-D evolvable cellular neural networks, a 32-bit genetic rule is included in each cell because von Neumann neighborhood are used. Here, two test patterns, 'Checker Board' (16x16) and 'Butterfly' (32x32) are applied in the evolving and test phases. Figure 9 shows the test phases in the 1190th configuration of 'Checker Board' in which the cell states, with the fixed genes, converge to the desirable bit-pattern within 20 steps. The evolved genes show a symmetric pattern inside the grid structure. For the sake of evaluation, the entropy values at time, t = 100, are also compared between two test phases in the 0th and the 1190th configurations of a 2-D evolvable cellular neural network as shown in Figure 10. As in the previous case, the spatial entropy is reduced in the later configuration due to self-organization as the evolving phases proceed. Note that as the time increases, the latter spatial entropy decreases to zero with the 1190th configuration, but the former does not. Finally, in Figure 11, the genetic rules of a complex bit-pattern 'Butterfly' are evolved and tested under 10% bit-reversal noise. Remarkably, the grades of the genetic rules in the grid structure show resemblance with the prescribed bit-patterns.
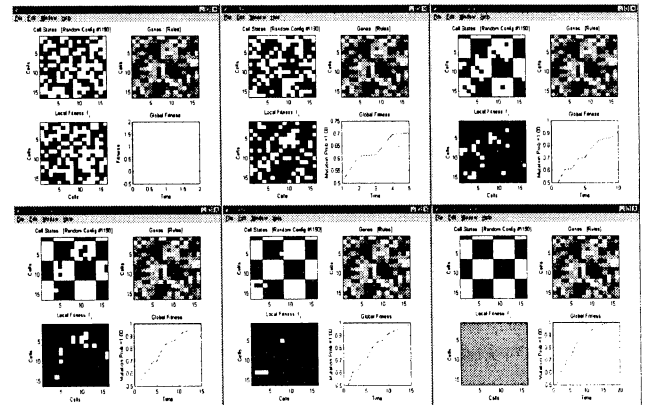
Figure 9. Test Phase of 1190th Configuration in 2-D Evolvable Cellular Neural Networks (Checker Board)

(a) Cell Entropy (0ᵗʰ Config.) (b) Cell Entropy (1190ᵗʰ Config.)

Figure 10. Spatial Entropy of 2-D Evolvable Cellular Neural Networks
(at t=100, Checker Board)

Figure 11. Test Phase of the 289ᵗʰ Configuration in 2-D Evolvable
Cellular Neural Networks (Butterfly)

Figure 12. Core Processing Unit of Evolvable Cellular Neural Network
(1-D with 3 Neighborhood)
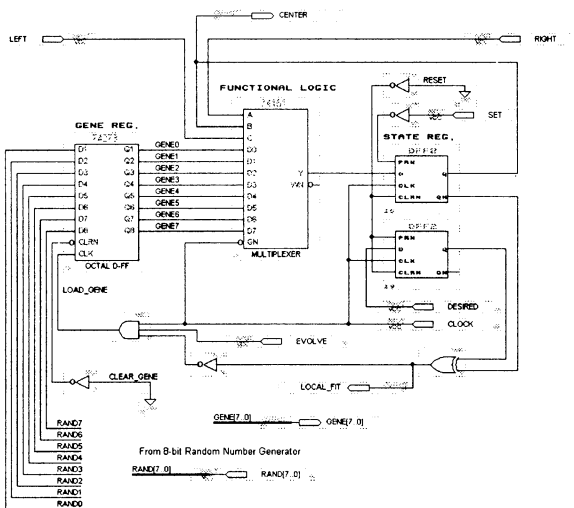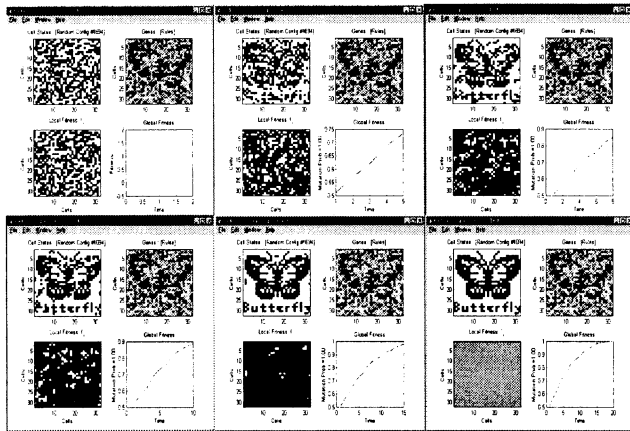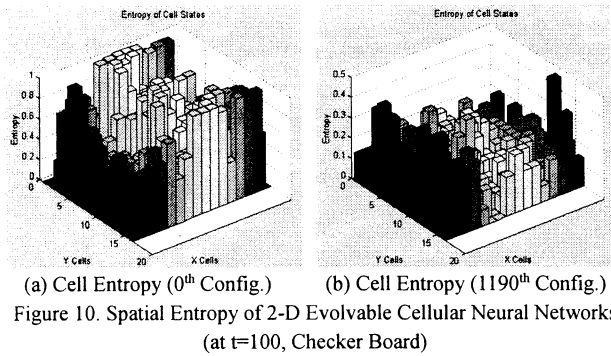
◆ **Evolvable Hardware : Implementation**

We have designed and successfully tested the core architecture of a 1-D evolvable cellular neural network as shown in Figure 12. The genetic rules and the cell state are stored in octal D flipflops, and a D flipflop, respectively. Also, the inputs of the octal D flipflops are linked to and 8-bit random number generator. The logical function unit simply consists of an 8-to-1 multiplexer whose inputs are connected to the outputs of octal D flipflops. Moreover, the local fitness value resulted from the exclusive NOR circuit is fed to the positive-edge-triggerred enable input of the gene memory.

III. Conclusions and Discussion

A new paradigm of cellular classifiers, *evolvable cellular neural networks*, is proposed, and an online evolutionary algorithm is developed. Initially, the genes are evolved, and then, both evolving and test phases take turns in order to separate randomly disturbed bit-string vectors into the nearest bit-patterns. Thus, evolvable neural networks are derived in terms of non-uniform cellular automata where the nonlinear function units are replaced with the nonlinear logical and-or gates. The core processing elements refer to the neighborhood of each cell and perform self-organization of the genetic rules that eventually classify the desirable bit-patterns due to redundancies in the genetic rules. Moreover, an evolvable hardware of the proposed neuronic architecture is implemented and successfully tested. It is believed that the mechanism of evolvable cellular neuronic systems make a new promising tool for classification of bit-string information.

References

[1] C. G. Langton, "Life at the edge of chaos", in *Artificial Life II*, vol. X of *SFI Studies in the Science of Complexity*, (eds. C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen), Massachusetts: Addison-Wesley, pp.41-91, 1992

[2] J. von Neumann, *Theory of Self-Reproducing Automata*, (ed. A. W. Burks), Illinois: Univ. of Illinois Press, 1966

[3] M. Gardner, "On cellular automata, self-reproduction, the Garden of Eden and the game of 'Life'", Scientific American, Feb. 1971

[4] E. F. Codd, *Cellular Automata*, New York: Academic Press, 1968

[5] H. deGaris, "Cam-Brain, ATR's billion neuron artificial brain project: A three year progress report", in Proceedings of IEEE 3ʳᵈ Int. Conf. on Evolutionary Computation (ICEC'96), pp.886-891, 1996

[6] S. Wolfram, *Cellular Automata and Complexity: collected papers*, New York: Addison-Wesley, 1994

[7] S. Levy, *Artificial Life: The Quest for a New Creation*, Random House, 1992

[8] M. Mitchell, J. P. Crutchfield, and P. T. Hraber, "Evolving cellular automata to perform computations: Mechanisms and impediments", Physica D, vol.75, pp.361-391, 1994

[9] M. Sipper, *Evolution of Parallel Cellular Machines: The Cellular Programming Approach*, New York: Springer-Verlag, 1997

[10] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities", in Proceedings of National Academic Sciences USA, Biophysics, vol.79, pp.2554-2558, Apr. 1982

[11] B. Kosko, "Bidirectional Associative Memories", IEEE Trans. on Systems, Man, and Cybernetics, vol.18, no.1, pp.49-60, Jan./Feb. 1988

[12] H. Kang, "Multi-layer associative neural networks: storage capacity vs. perfect recall", IEEE Trans. on Neural Networks, vol.5, no.5, pp.812-822, Sep. 1994

[13] C. E. Shannon, "The mathematical theory of communication", The Bell System Technical Journal, vol.27, pp.379-423, pp.623-656, 1948.

[14] Z. Navabi, *VHDL Analysis and Modeling of Digital Systems*, New York: McGraw-Hill, Inc., 1993

Applications of Artificial Life to Developing Robot and Softbot*

Sung-Bae Cho
Department of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
E-mail: sbcho@csai.yonsei.ac.kr

Abstract

Adaptation gives rise to a kind of complexity that greatly hinders our attempts to solve some of the most important problems currently posed by our world. Recently, there is an attempt to build a complex adaptive system, which is rich in autonomy and creativity, with the ideas and methodologies of Artificial Life (A-life). This paper presents the concepts and methodologies of A-life, and shows two typical applications developed based on them. These systems cannot only develop new functionality spontaneously but also grow and evolve its own structure autonomously. They have been applied to controlling a mobile robot and developing adaptive agents on the WWW.

Key words: Artificial life, complex adaptive systems, robot, softbot.

1 Introduction

Intelligent systems can adaptively estimate continuous functions from data without specifying mathematically how outputs depend on inputs. System behavior is called *intelligent* if the system emits appropriate problem-solving responses when faced with problem stimuli. Recently, some researchers have tried to synthesize intelligent systems by using Artificial Life (A-life) technologies.

A-life research aims at studying man-made systems exhibiting behaviors characteristic of natural living systems. It complements the traditional biological sciences concerned with the analysis of living organisms by attempting to synthesize life-like behaviors within computers; extending the empirical foundation upon which biology is based from *life-as-we-know-it* to a larger picture of *life-as-it-could-be* [4]. The essential features of A-life models are as follows:

- They work with populations of simple programs,

where no single program directs all of the other programs.

- Each program details the way in which a simple entity reacts to local situations in its environment, including encounters with other entities.

- There are no rules in the system that dictate global behavior, and higher behavior is therefore emergent.

It is the concept of emergent property that highlights the nature of the A-life research. Emergent property is exhibited by a collection of interacting entities whose global behavior cannot be reduced to a simple aggregate of the individual contributions of the entities. Conventional methods in artificial intelligence have difficulty to reveal and explain the emergent properties, because they are generally reductionist that decompose a system into constituent subsystems and then study them in isolation according to top-down approach.

On the other hand, A-life adopts bottom-up approach which starts with a collection of entities exhibiting simple and well-understood behavior and synthesizes more complex systems. There are a lot of technologies in the A-life research such as cellular automata, Lindenmayer system, genetic algorithm, neural networks, and so on, but the key idea is the evolutionary algorithm. In this sense, a practical goal of A-life can be redefined as finding a mechanism for an evolutionary process to be used in the automatic design and creation of artifacts.

A genetic algorithm, one of the evolutionary algorithms, is a model of machine learning derived by the procedure of evolution in nature. This is performed by creating the population of individuals that are represented by chromosomes. The chromosome is the string that can be thought of as the human genes. The individuals in the population go through the evolution. This procedure takes the evolutionary procedure in which different individuals compete for resources in the environment. Better individuals are more likely to survive, and propagate their genetic material to offsprings.

2 Application to Mobile Robot

There have been several attempts to develop an artificial brain with engineering techniques. Among them CAM-Brain develops neural networks based on cellular automata by evolution. In particular, due to their features Cellular Automata (CA) can be evolved very quickly on parallel hardware such as CAM-8 at MIT, or CBM at ATR [3].

Evolutionary Engineering (EE) is an approach to evolve neural network modules with particular functions to develop artificial brain. It has been extensively exploited to apply each neural network module to a specific problem. We have attempted to evolve a module of CAM-Brain for the problem of robot control, especially Khepera. A simulation indicates that an appropriate neural architecture emerges to make Khepera simulator to navigate the given environment without bumping against walls and obstacles. This section presents the power of the model based on A-life technology by analyzing the robot behaviors and corresponding neural networks evolved.

CAM-Brain is an evolved neural network based on CA. This paper uses one of the CAM-Brain models, CoDi model, and the process of developing neural networks and signaling among neurons. This process consists of two phases. One is growth phase that makes the structure of neural network. The other is signaling phase that sends and receives signals among neurons.

2.1 Method

Applying CAM-Brain to a mobile robot control [2] requires the following process. Neural network structure is made in growth phase. In signaling phase, sensor values from Khepera simulator [1] are used as inputs to CAM-Brain. CAM-Brain transmits signal from input to output cells. As the output values of CAM-Brain are inputed to Khepera simulator, the robot moves. When the robot bumps against the obstacle or reaches the goal, its fitness is computed. Chromosomes are reproduced in proportion to this evaluation.

There are a couple of problems in applying the model to controlling robot. Because CAM-Brain cannot perfectly utilize activation values of robot sensors the activated range of input neuron is made differently according to the value. In addition, because delay time is needed until CAM-Brain makes output value, we execute dummy signaling phase for some duration until the signals started from input cells arrive at output cells. It makes the robot to react appropriately in several situations [1].
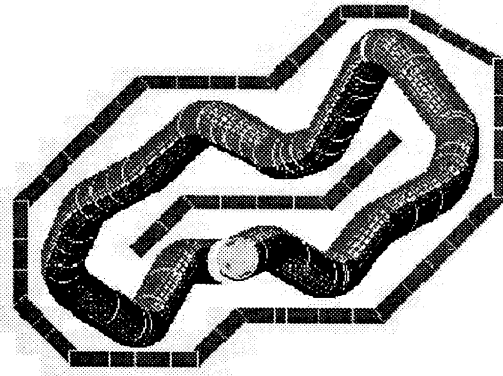
Figure 1: The trajectory of a robot.

2.2 Results

The robot controller evolves in 5×5×5 CA space to facilitate an easy analysis. After the 21st generation, individuals that have fitness value one keep on appearing. Fig. 1 shows the trajectory of a successful robot. This is less smooth than that obtained in our previous work, but this robot controller has smaller size, which makes the analysis easier.

Fig. 2 shows an architecture of the neural network evolved. Dotted arrow represents inhibitory connection, and lined arrow represents excitatory connection. This has been obtained by tracing the activation values of each neuron. The number of neurons is 12, but neurons 8, 11 and 12 are not functional because these neurons are not in the path of input to output neurons. Neurons 2 and 10 are output neurons to produce the velocity of left and right wheels, and neurons 3, 4, 5 and 6 are input neurons. Neuron 3 is for front sensor of the robot, neurons 5 and 6 are for left sensor of the robot, and neuron 4 is for right sensor of the robot.

The architecture of controller has direct connections from input to output neurons. These connections play a role for turning left and right: If neuron 5 has high activation value (which means no obstacle at the left side of the robot because we scale sensor value of the robot inversely), neuron 10 (right wheel) produces positive signal (because neuron 5 fires excitatory signal to neuron 10). If neuron 4 has low activation value (obstacle at the right side of the robot), neuron 2 (left wheel) produces 0 (because neuron 5 cannot fire neuron 2). By the way, the velocity of each wheel is decided according to the value of the output neurons. It becomes 5 (if the output value is positive), −5 (if negative) or 0 (otherwise). The values of the output neuron make the robot to turn left. Similarly, the robot can turn right with the addition of neuron 7.
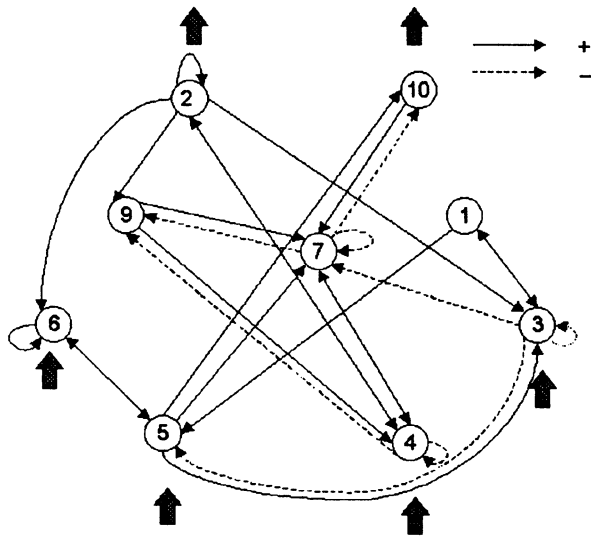
Figure 2: Schematic diagram of evolved neural network.

Table 1: Overall algorithm

```
Initialize agents;
Obtain queries from user;
while (there is an alive agent) {
    Get document Dₐ pointed by current agent;
    Pick an agent a randomly;
    Select a link and fetch selected document Dₐ';
    Compute the relevancy of document Dₐ';
    Update energy (Eₐ) according to the document
        relevancy;
    if (Eₐ > ε)
        Set parent and offspring's genotype
            appropriately;
        Mutate offspring's genotype;
    else if (Eₐ < 0)
        Kill agent a;
}
Update user profile;
```

3 Application to Softbot

WWW has a large distributed collection of documents, which can be added, deleted or modified dynamically. Moreover, the document style is various. It takes much time and effort for users to search the Web in this environment. Due to these reasons, several search agents have been developed and investigated.

Conventional search agents retrieving information on the Web are mainly devised for static and non-distributed environments. With these agents, end user gives queries to the server that maintains index files to get relevant document lists. User's requests are processed through the use of index files, which are made and updated by off-line robot agents that collect and analyze the documents. Because of its fast response time these search agents are prevalent now, but they have several limitations. First, they cannot cope with dynamic change of documents. Second, they can abstract away important data by incorrect indexing, and missing relations between documents. Third, they cannot reflect on the user's preferences or habits. To overcome these limitations, a new method is required to replace the index-based robot agents.

Our A-life agent is very similar to Infospider that was originally proposed by Menczer [5]. It has a population of on-line agents that search documents by deciding own actions locally. Each agent in a population can reproduce the offspring, or disappear, according to the relevancy of the documents retrieved by the agent. The population of agents converges to optimal states through evolution. However, if we incorporate user's preference, we can provide more accurate information quickly and personalize the agents for user. By updat-

ing user profile at each queries, we can reflect user's preferences. A-life agents maintain their competence by adapting to the user's preference that may change over time.

3.1 Method

The authors of web document tend to classify the documents according to subjects and connect them in related topic. This tendency results in semantic topology, which defines the correlation of documents. If some documents are relevant to users, the links in current document are also highly relevant to users. Also, the links close to meaningful keywords are more probable than other links. The artificial life agents can reduce the search space by using this property. It has a population of multiple retrieval agents. The energy of each agent in population is increased or decreased by relevance of the document retrieved by agent itself. This method uses genetic algorithm based on local selection. The algorithm is shown in Table 1.

3.2 Results

In order to provide a fair and consistent evaluation of the system's performance, we have restricted search space to the local machine, instead of real Web. We have collected a number of HTML pages with various topics, classified the pages according to the subjects, and put them in different directories. The initial user profile is composed of the top directories of local machine. The initial population size of agents depends on the number of URLs in user profile. We have compared the A-life agents with Breadth First Search (BFS) and

Figure 3: Hit rate.

Figure 4: Response time for the queries in the same category.
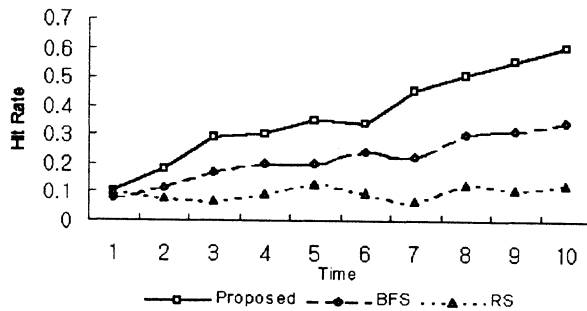
Random Search agents. BFS searches every documents exhaustively, while the A-life agents can search documents selectively.

The initial population is composed of 10 agents. The population size has no limitation in run time. The constant threshold ε is set to 0.4. Agent whose energy is greater than ε can reproduce offspring. Initial agent's energy is set to $\varepsilon/2$. The agent uses the network resource, which means loss of energy. This loss of energy is called *expense*, and set to 0.1. The process is influenced by the *expense* value. If we increase the *expense* value, the agents have less chance to search further. Irrelevant agents may disappear quickly, and there is some possibility that even some relevant agents can disappear without searching the regions sufficiently. We select the *expense* value with trial and errors.

The most important property of the A-life agents can discard the useless agents irrelevant to the user's preference effectively. Fig. 3 shows the hit rate of relevant documents. In the beginning, the performance is not better than other search methods, but the performance of the proposed agents is improved rapidly. This result implies that each agent can cut irrelevant paths of documents effectively. The action of each agent goes toward relevant document paths gradually. By using this property, A-life agents can reduce the access of irrelevant documents.

We have tested the performance improvement in case the user gives queries in the same category. For each query, the user profile is updated according to document relevancy, and the agents reflect the user's preference adaptively. As the user gives the queries in the same category, the proposed agents improve the response time according to the queries. Fig. 4 shows the results on the two tasks. In task 1, a sequence of queries is provided as computer, artificial intelligence, neural network, agent, evolution, user feedback, retrieval, and search. In task 2, the sequence is computer, DSSSL, SGML, grove, property set, repository, and database. Initial response time is not good, but as the queries are
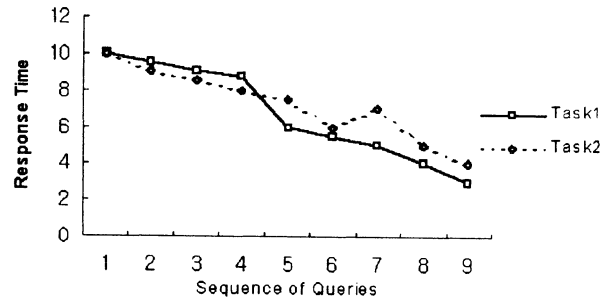
given repeatedly, we can see the improvement of the response time for the two tasks.

4 Concluding Remarks

This paper introduces the key concept of A-life technology and shows the potential with the applications such as controlling a mobile robot and developing adaptive agents on the WWW. While artificial intelligence uses the technology of computation as a model of intelligence, A-life is attempting to develop a new computational paradigm based on biological processes.

References

[1] S.-B. Cho, G.-B. Song, J.-H. Lee and S.-I. Lee, "Evolving CAM-Brain to control a mobile robot," *Proc. Int. Conf. on Artificial Life and Robotics*, pp. 271–274, 1998.

[2] D. Floreano and F. Mondada, "Evolution of homing navigation in a real mobile robot," *IEEE Trans. Systems, Man, and Cybernetics*, 26 (3), pp. 396–407, 1996.

[3] H. de Garis, "CAM-Brain : ATR's billion neuron artificial brain project: A three year progress report," *Proc. Int. Conf. on Evolutionary Computation*, pp. 886–891, 1996.

[4] C.G. Langton, "Artificial life," *Artificial Life*, Addison-Wesley Publishing Company, pp. 1–48, 1989.

[5] F. Menczer, "ARACHNID: Adaptive retrieval agents choosing heuristic neighborhoods for information discovery," *Proc. 14th Int. Conf. on Machine Learning*, 1997.

Cooperative Behavior of Collective Autonomous Mobile Robots Based on Online Learning and Evolution

Dong-Wook Lee, Ho-Byung Chun, and Kwee-Bo Sim

School of Electrical and Electronics Engineering, Chung-Ang University
221, Huksuk-Dong, Dongjak-Ku, Seoul 156-756, Korea
dwlee@ms.cau.ac.kr, courtil@jupiter.cie.cau.ac.kr, kbsim@cau.ac.kr

Abstract

In distributed autonomous robotic systems, each robot must behaves by itself according to its states and environments, and if necessary, must cooperates with other robots in order to carry out a given task. Therefore it is essential that each robot has both learning and evolution ability to adapt the dynamic environments. In this paper, the new learning and evolution method based on reinforcement learning and distributed genetic algorithms is proposed for behavior learning and evolution of collective autonomous mobile robots. Reinforcement learning having delayed reward is still useful even though when there is no immediate reward. And each robot can improve its behavior ability by distributed genetic algorithm exchanging the chromosome acquired under different environments. Specially, in order to improve the performance of evolution, selective crossover using the characteristic of reinforcement learning is adopted in this paper. We verify the effectiveness of the proposed method by applying it to cooperative search problem.

Keywords: distributed autonomous robotic system (DARS), reinforcement learning, distributed genetic algorithm, cooperative behavior

I . Introduction

In recent years, the research on Distributed Autonomous Robotic Systems(DARS) has been becoming active because of expansion of the fields in the robotic application. DARS has several merits compared with centralized robotic system. Its most significant merit is that each robot perceive its environments such as object and the other robot's behavior etc., and they determine their behavior independently, and cooperate with other robots in order to perform the given tasks very well.

The effect of DARS is revealed as group behavior and cooperative behavior. The group behavior can be realized through only sensing its environment. But in the case of cooperative behavior, additionally high reasoning ability is required to predict other robot's behavior. It can be achieved by communication among robots. Each robot exchanges their state and information through communication, and they can easily cooperate in given tasks. In DARS, each robot doesn't necessarily have to know all the information that other robots have, they only perceive around

their environments. In order to decide their behavior in the system, they communicate with other robots and renew the information to their environments.

Another merit of DARS is flexibility and robustness. That is, DARS can apply a robot to the several tasks by making the robot to change only its behavior rule according to given tasks. It doesn't affect performance of the system although a few robots are out of order. Also even if the system size is large, the complexity of the system doesn't increase.

Generally in distributed system, it is difficult that a robot decides best action by itself for cooperative behavior because the environments change dynamically. For this reason, the new approach based on artificial life, which is modeled to natural living creature, is expected as a solution of unpredictable and complicated problem. As a result of this approach, there are behavior-based robot proposed by Brooks[1] and collective behaviors realized by Mataric[2]. In recent years, neural networks, fuzzy system, evolutionary algorithms, and immune systems or these fusion technique etc. are gradually applied to this problem.

Artificial life robot(we call it agent), authors think, has two features as follows; (1) An agent must immediately cope with an unpredictable problem and posses learning and adaptation ability instead of prefixed perfect plan in solving problem. (2) It appears emergent behavior by interaction among agents and environment. Also, an agent learns by watching and imitating other agents in nature. This is very practical way for an agent to gain additional knowledge and it has been the basis for some very successful robot learning programs. However, perception remains as a big difficulty. Most robots don't have sufficiently advanced perception to be able to sense what other robots are doing and whether they are succeeding.

From this point of view, we tried realizing a robot with artificial life using distributed genetic algorithms in this paper. In order to achieve cooperative behavior for given tasks, each robot has basic ability to sense its environments and communicate with each other. A robot learns proper behavior by sensing and action to the reinforcement signals, and evolute behavior strategies with exchanging information between robots by communication. This evolution looks as if a living creature learns by watching and imitation in nature.

II. Sensing and Communication System

To execute complicated and sophisticated tasks by cooperative behaviors, it is essential to use communication in DARS. In general communication can be classified into global and local one. A global communication is effective for small number of robots. However, when the number of robots goes on increasing, this becomes difficult to be realized because of limited communication capacity and increasing amount of information to handle. Also the problems such as communication interference and improper message transmission occur. Thus we adopt a local communication system in which each robot transmits information locally because it is possible to prevent not only overflow of information but also complexity of communication.

In this paper, we use infrared sensor for sensing and communication. A robot can sense distance to other robots and obstacle, and it transmits information infrared pulses in sequence. In case that each robot face each other, communication between robots are carried out. In addition to this, a robot has color sensor that can distinguish object from obstacle in front of robot.

Each robot diffuses information around with sign-board model. If a robot encounters another robot whose fitness is higher than it, this two robots communicate each other by message-passing model. Figure 1 shows sensing and local communication system of autonomous mobile robot[3].
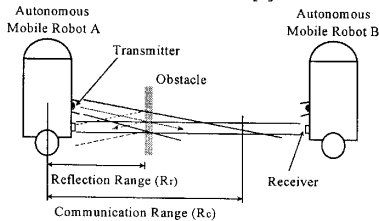
Fig. 1 Sensing and local communication system of autonomous mobile robot.

III. Concept of Proposed System
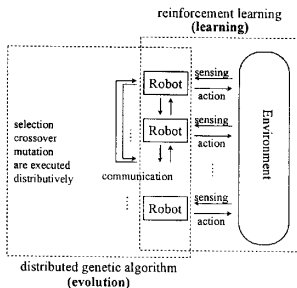
Fig. 2 Block diagram of proposed system.

The diagram of proposed system in this paper is shown in figure 2. Each robot has several state-action rules, which means that action follows state, as a table form. According to the reward or penalty given by result of action, each robot renews the rule just as Q-learning. When a robot encounters more excellent robot than itself, this robot obtains the rule through communication, and reproduces a new rule using genetic operations.

IV. Reinforcement Learning-based Behavior Learning

4.1 Reinforcement learning

The objective of reinforcement learning is to maximize the rewards that a learning agent receives for the purpose of improving its behaviors through the interaction between itself and an environment using a reinforcement signal[4].

And besides, the Q-learning is developed as one method of model-free reinforcement learning based on stochastic dynamic programming. It provides robots with the capability of learning to act optimally in a markovian environment. We assume that the robot can discriminate the set S of distinct world states, and can take the set A of actions on the world. A simple Q-learning algorithm used here is shown in Figure 3[5].

1. Initialize $Q(s,a)$ to 0 for all state s and action a.
2. Perceives current state s.
3. Choose an action a according to the state-action rules.
4. Carry out action a in the environment. Let the next state be s' and immediate reward be r.
5. Update state-action rule from s, a, s', and r,

$$Q_{t+1}(s, a) = (1 - \alpha_t)Q_t(s, a) + \alpha_t(r + \gamma \max_{a' \in A} Q_t(s', a'))$$

where α_t is a learning rate and γ is a fixed decreasing factor between 0 and 1.
6. return to 2.

Fig. 3 Simple Q-learning algorithm.

4.2 Realization of cooperative behavior

Cooperative search and delivery is feasible cooperative work by multiple robots in DARS. As examples in real world, investigation of underwater for collecting resources and marine products, development of space, collecting objects in wide area, and works in manufacturing system etc. are exist. In the future, as technology of micro robot advances, applying field of DARS will be wider.

In this paper, in order to verify the effectiveness of the proposed system, we determine the objective of the system as collecting objects and delivery by robots. So a robot finds objects and cooperative delivery from avoiding collision with obstacle and other robots in given environment.

4.3 States(situations)

Robot has one color sensor which is distinguish a given task(objects) from several obstacle, and eight distance sensor.

Table 1 is possible states of robot. Total states are $81(3 \times 3 \times 3 \times 3)$.

Table 1. Robot's states.

Sensor	Forward (S0)	Right (S1,S2,S3)	Left (S5,S6,S7)	Rear (S4)
State	Nothing Object Obstacle or Robot	Nothing Object Obstacle or Robot	Nothing Object Obstacle or Robot	Nothing Object Obstacle or Robot

4.4 Actions

Basic behaviors of robot to act are as follows.

- Random move(turn arbitrary direction and go forward)
- Turn right
- Turn left
- Turn 180 degree
- Go to target(object, robot, obstacle etc.)

In this system, it is impossible to predict next state s', we used modified Q-learning in figure 4.

---

1. Initialize $Q(s,a)$ to 0.2 for all state s and action a.
2. Perceive current state s.
3. Choose an action a in proportion to probability $P(a_i)$

$$P(a_i) = \frac{Q_i^2}{\sum_{j=0}^{N-1} Q_j^2}$$

4. Carry out action a in the environment. And calculate delayed reward r_d $(-1 \leq r_d \leq 1)$.
5. Update Q-value from s, a and r_d,

$$Q_{t+1}(s, a) = (1 - \alpha_t)Q_t(s, a) + \alpha_t(0.5 r_d + 0.5)$$

where α_t is learning rate, and $Q(s,a)$ has real value between 0 to 1.
Even though r_d isn't calculated immediately, Q-value is updated as a whole to prior behavior, once r_d is calculated.
6. return to 2.

Fig. 4 Modified Q-learning.

V. Evolution of Behavior Strategy Based on Distributed Genetic Algorithm

5.1 Distributed genetic algorithm

A distributed genetic algorithm can be classified into three-execution method. The first approach method is that population is divided as part and each part is put together by periods in evolution[6]. The second is that each agent has one chromosome, and evolute mutually by communication in distributed system[7]. The third is that chromosome is divided as part, and each part of chromosome evolute respectively, finally they are put together[8]. In this paper, we use second method for robots to evolve in DARS.

In simple genetic algorithms, fitness calculation, selection, reproduction(crossover and mutation) are executed as a whole. But in distributed genetic algorithm, these are executed separately. So each robot has ability of fitness calculation by reinforcement learning and ability of selection and reproduction by communication. In this processing, a robot improves his performance by adding other robot's chromosome, which is obtained different environment, to self-chromosome. This is processing of cooperative learning(or evolution) of multiple robots.

◆ Chromosome
Robot's chromosome is Q-values of Q-table. So chromosome is composed of series of real value.

◆ Fitness function
A robot calculates its fitness by counting reward and penalty signal which was paid during last T_{eval} times. This value is based on selection.

fitness $= \alpha \times$ # of reward $- \beta \times$ # of penalty (1)

◆ Selection
When a robot passes T_{eval} times. It can be selected by other robots. This time is minimum evaluation time of fitness. If robot A encounters robot B and robot B's fitness is higher than A's. Then robot A selects robot B. And robot A receives robot B's chromosome and reproduces self-chromosome.

◆ Reproduction(crossover)
Because a chromosome is a series of Q-valuse of Q-table. One state and its following actions is one gene. So total number of gene is 81(because total states are $81=3^4$). In this paper, we adopt modified uniform crossover. Proposed crossover is as follows.

We represent the chromosome of robot as a pair of x(gene) and q(learning times). Then the chromosomes of two parents are represented as (2).

$$(\vec{x}^1, \vec{q}^1) = ((x_1^1, \cdots, x_n^1), (q_1^1, \cdots, q_n^1))$$
$$(\vec{x}^2, \vec{q}^2) = ((x_1^2, \cdots, x_n^2), (q_1^2, \cdots, q_n^2)) \quad (2)$$

where, n is total number of gene.
New offspring, which is generated by proposed crossover method, is represented as equation (3).

$$(\vec{x}, \vec{q}) = ((x_1^{s_1}, \cdots, x_n^{s_n}), (q_1^{s_1}, \cdots, q_n^{s_n})) \quad (3)$$

where, $s_i = \begin{cases} 1 & p_i < \dfrac{q_i^1}{q_i^1 + q_i^2}, i = 0 \cdots n, \\ 2 & else \end{cases}$

p_i = random number from 0 to 1.

So, offspring's chromosome is inherited from parents

1 and parents 2 according to the learning times(q).

By this selection and crossover, a robot obtains learning data of unknown environment that it hasn't been.

VI. Simulation Results

6.1 Simulation condition and environment

- Problem : Cooperative search and delivery.
- Number of robot : 25
- Number of object : 500
- Number of obstacle : 100
- Evaluation time(T_{eval}) : 500 sec.
- Work area : 20×20m(robot size: 5×5cm)
- Communication radius : 75cm
- Sensing range : 32.5cm
- Object and obstacle are spread out in working area.
- First, all robots spread out in order to establish and maintain some minimum inter-robot distance
- Simulation condition
  - ① Without learning and evolution
  - ② With only learning
  - ③ With learning and evolution
    - ● $\alpha = \beta = 5$
    - ⓐ $\alpha \gg \beta$
    - ⓑ $\beta \gg \alpha$

6.2 Simulation results
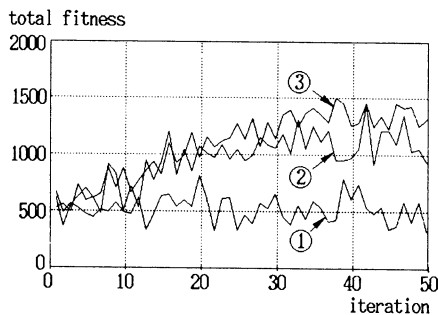
Fig. 5 Relationship between total fitness variation and iteration numbers($\alpha, \beta = 5$).

Figure 5 shows total fitness variation of robots as iteration time increases(1 iteration is 1000 sec.. Fitness is calculated and objects are fulfilled every iteration).
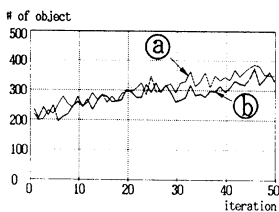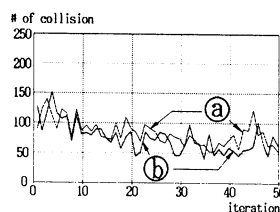
Fig. 6   Fig. 7

Fig. 6 Relationship between the number of obtained object and iteration.

Fig. 7 Relationship between the number of collision times and iteration.

VII. Conclusion

In this paper, we introduce reinforcement learning and distributed genetic algorithm for behavior learning and evolution of multiple robots in DARS. Each robot decided its action by perception of environment and learned proper behavior by reinforcement learning. And a robot was evoluted cooperatively using communication. By computer simulation, we knew that proposed method (reinforcement learning and distributed genetic algorithm) is effective in DARS.

In a system which is composed of multiple mobile robots, it is difficult to make behaviors rule considering with dynamic environment. In recent years, many researchers are interested in artificial life approach instead of conventional AI approach. Especially, neural networks (include reinforcement learning), genetic algorithms, fuzzy system, and the fusion of these are paid attention to. In this paper, we realize cooperative behavior by making control structure which proper behaviors can emerge and grow, instead of making perfect program.

In the future, as technology of autonomous mobile robot and micro robot advances, it is very important to be applied proposed method at DARS.

References

[1] R.A. Brooks, "Behavior Humanoid Robotics," *Proc. of IROS 96*, pp. 1-8, 1996.

[2] M.J. Mataric, "Designing Emergent Behaviors: From Local Interactions to Collective Intelligence," *Proc. of 2nd Intr. Conf. on Simulation of Adaptive Behavior*, pp. 432-441, 1993.

[3] D.J. Kim, D.W. Lee, K.B. Sim, "Development of Communication System for Cooperative Behavior in Collective Autonomous Mobile Robots," *Proc. of 2nd ASian Control Conference*, vol. 2, pp. 615-618, 1997. 7.

[4] L.P. Kaelbling, "On Reinforcement Learning for Robotics," *Proc. of IROS 96*, pp. 1319-1320, 1996.

[5] C.J.C.H Watkins, P. Dayan, Technical Note : "Q-Learning," *Machine Learning*, vol. 8, pp. 279- 292, 1992.

[6] A. Loraschi et. al., "Distributed Genetic Algorithms with An Application to Portfolio Selection Problems," *Proc. of Intr. Conf. on Artificial Neural Nets and Genetic Algorithms*, pp. 384-387, 1995.

[7] E. Horiuchi, K. Tani, "Behavior Learning of Group of Mobile Robots with a Distributable Genetic Algorithm," *J of RSJ*(Japanese), vol. 11, no. 8, pp. 1212- 1219, 1993.

[8] D.W. Lee, K.B. Sim, "Evolution of Behavior in Collective Autonomous Mobile Robots using Distributed Genetic Algorithms," *Proc. of JCEANF*(Korean), pp. 34-37, 1996. 10.

[9] T. FuKuda, T. Ueyama, *Cellular Robotics and Micro Robotic System*, World Scientific, 1994.

DAYDREAMING ROBOTS

Syed I. Ahson
King Saud University
Computer Engineering Department
Riyadh 11543, PB 51178, Saudi Arabia
ahson@ccis.ksu.edu.sa

Keywords: daydreaming, connectionist models, robots, emotions, neural networks

Abstract

The inner experience of " daydreaming" in humans has been studied by psychologists and philosophers since the emergence of the cognitive movement in the sixties. In recent years daydreaming has attracted the attention of computer scientists and roboticists for exploring computational theories of mind and for designing imaginative robots that are provided with the ability of learning from the past ,planning for the future, and making intelligent decisions. This paper presents a framework for a connectionist model of daydreaming in robots. The model is based on a recently proposed relational theory of mind , and uses neural networks as the building blocks that together generate daydream scenarios.

1. Introduction

William James [1] coined the famous metaphor " stream of consciousness" to convey the idea that while consciousness itself seems to be continuos , within each personal consciousness the contents are always changing. Before building a robot with human-like mind it is necessary to study the "trains of thought" that occurs in humans. Such spontaneous or controlled sequence of thought has been termed "Daydreaming". Dennet writes in his book[2] : " There are robots that can move around and manipulate things as adeptly as spiders. Could a more complicated robot feel pain , and worry about its future , the way a person can ?" Later on he writes "we know that people the world over have the same likes and dislikes ,hopes and fears .We know that they enjoy recollecting favorite events in their lives .We know that they all have rich "episodes of waking fantasy" , in which they re-arrange and revise the details deliberately ..we know that they can be reminded of an aroma or a melody of a specific event in their lives , and that they often talk to themselves silently ..".

Mueller [3] has implemented a DAYDREAMER program for exploring a computational theory of daydreams. He has proposed daydreaming as a useful capability for computers and a necessary capability for intelligent robots. He argues that daydreaming : a)improves efficiency by exploiting idle-time for learning from past experience and for planning for the future, b)provides option for analogical reasoning using task scenarios, and c)prevents emotion from getting out of control. Mueller has used the symbolic AI paradigm for his DAYDREAMER.It is now well-known that the brain is fundamentally different from a computer serially manipulating symbols. Although his model can be used as control mechanism for autonomous robots , it cannot produce rich cognitive content intrinsic to imagination. Surely, we need a connectionist model to satisfactorily represent daydreaming.

The neural network modeling techniques have come a long way since McCullock and Pitts [4] first proposed their neuron model. The state of the art of modeling has now advanced from lower-order pattern processor to higher-order cognitive brain functions. There has been a concerted effort in recent years toward the development of models related to consciousness[5-8]. In this paper ,we present a framework for a connectionist model for generating daydreams in computers. This is a step toward constructing robots capable of deciding what to do next based on their motivating emotions. The robot can take actions based on planned "virtual actions". For the purpose of modeling daydreams in robots we limit ourselves to controlled "train of thought" which they generate and manipulate voluntarily , rather than spatialization or spontaneous thoughts which occur in visual imagery. Such conscious thoughts involve emotional states , recalled past events, imagined variations on past events and imagined future events. We hope our ideas will motivate many researchers who are trying to build intelligent robots having some of the intellectual capabilities found in humans.

2. Imagination and Daydreaming in Humans

Daydreaming is one specific use that we can make of our mental capacity for imagination and mental imagery .Some people consider daydreaming as thoughts that involve self-amusement(fantasies, wish-fulfillment) or self-gratification rather than any serious and productive task. Such daydreaming express our emotional needs and

desires and may serve a useful psychological function, for example relieving stress. Most people apply the term " daydreaming" to thoughts that are not directly related to the task at hand in the physical or social environment. The term usually carries the connotations of idleness and self-indulgence. But often daydreaming serves valuable function , and it is far from being a waste of time[9].Daydreaming is a kind of imagination which involves exploration of possible actions we might take in certain possible future circumstances. Daydreaming can have practical value if the thoughts are task-related rather than on entertainment or worrying where there is no task at hand or the task is easy. Daydreaming can be used for generating solutions to life problems and for planning and rehearsing for the future. In this the sequence of daydream thoughts are not entirely random, rather they are operant, realistic and well-integrated[10].

Daydreaming is unlike most human activities since it is not manifested externally. There is no apparent output when a person is daydreaming. It is externally visible , partially only through behavior. The presence of intelligent behavior in humans is visible in their competence to adapt to changing circumstances and challenges. Behavior is influenced by knowledge acquired by them which they can utilize in a variety of situations. Imagined thoughts can represent objects and events not currently present and the knowledge gained in the past is used to deal with the present. Humans also predict the future , at least the short-term future. They can anticipate future needs, events and actions. They also reshape the past in the light of the present[5].In these activities of predicting the future and reshaping the past daydreaming plays an important role.

The existence of three "styles" of daydreaming has been documented in the literature [11].A Guilty-Dysphoric waking fantasy is dominated by negative emotions- guilt, fears, conflicts, hostilities. A Positive-Vivid type of waking fantasy involves positive emotions and vivid sensory imagery. Persons employ this type of daydreaming for constructive problem solving. The Anxious-Distractible fantasy is often intrusive , frightening ,or bizzare , alongwith difficulty in attending to tasks. Guilty-Dysphoric daydreams and terrifying nightmares are highly related phenomena. Striking similarities have been observed between dreams and psychotic experiences leading to the conception of hallucinatory psychosis as "waking dream". Vivid daydreaming is a normal human process unlike hallucinations in psychotics. In daydreaming the "ego" remains intact identifying itself with the imagined activity, while in "waking dreams" the ego becomes receptive to the imagined streams.

3.Consciousness –Based Robotics

Behavior-based approaches[12] have emphasized a direct perception-to-action mapping without modeling the physical world. The past few years has seen a flurry of research activity into the behavioral control of robots. Tasks are , however , limited to light- finding, wall-following, obstacle avoidance , and a combination of these. The activity in behavioral-robotics started with the work of Braitenberg [13] who proposed a series of hypothetical vehicles that starting from very elementary behavior " .. moved in the direction of creating behavior akin to thinking"[13]. He writes on p.62 of his excellent book "...thinking is different. It is a process that can go on for a long time ,as everyone who has done some conscious thinking knows .."Introducing his vehicle 13 named Foresight he writes " I am sure that most of you will not believe "aimless succession of images" is an accurate description of what goes on in your minds most of the time. You will not be impressed by our vehicles as long as there is no evidence of some purpose guiding their behavior and some direction in their thinking .These are virtues we are pleased to see in our children. Why should'nt we try to modify our brain children , the vehicles in this direction ?It won't be difficult in principle , and it means a lot to those philosophers who like to think that goal-directed behavior is the one property that gives living beings their very special status within the physical universe......We want our vehicles to be imaginative , but mainly realistic". Thus Braitenberg may be called the father of "Daydreaming Robots". His ideas have been implemented in practical robots , but these are far from being imaginative.

Another way of approaching behavioral robotics is through cybernetics as shown by Seymour Papert [14] in his book "The Children's Machine". Papert writes on p 182 – "The shift from AI to cybernetics widens the focus from prototypes of behavior with a primarily logical flavor to include prototypes with a more biological flavor. He describes about small children making wonderful devices that they describe as "dragons" ,"snakes" or "robots". The childrens' work is closer to "Artificial Life" than AI. Since this involves borrowing ideas from biology, psychology, control engineering and robotics , he uses the word "cybernetics" and "emergent programming" to describe these physical computational objects. He makes a case for cybernetics to be a "new" subject that will open new intellectual domains for children. A modern scientific outlook towards consciousness is now leading to a materialistic explanation of behavior based on a large set of interconnected "neural assemblies". Many theories of consciousness[5 – 8] have been proposed over the past few years. A more modern approach to building imaginative robots can now be based on the "science of

consciousness" as it evolves. People so far have rejected the idea that "robots can be conscious" because consciousness is still not well-understood. The variety of the contents of consciousness is enormous which include perception of objects and events, body sensations, memories of past events, imagined scenes, emotional feelings, and so on . For the daydreaming robots ,we have simplified our task by selecting to model a part of "inner experience"—controlled realistic thoughts. Although McCarthy[15] argues that robots can be provided with powerful introspective abilities and that a robot can observe its mental states, we can as a first step develop robots without the meta-level reflective capacities , concentrating only on how to achieve primary daydreaming in robots .For the purpose of this paper , we limit ourselves to a subset of contents of conscious thought consisting of some components of daydreaming : emotional states, recalled past events, imagined variations on past events and imagined future events. We rule out degenerative thought and visual imagery.

Will robots of the future be self-conscious ? Sloman [16] writes " ... not all animals can do all things. Not all humans can do them all : very young children , and people whose brains are either genetically inadequate or damaged by accident or disease , may lack some capabilities ." Writing about this issue Farthing [9] states : " Does everyone introspect ? To people who are in the habit of introspecting it may seem obvious that introspection is natural and everyone does it. But this is not necessarily the case .Poorly developed capacities for introspection , reflective thought , and self –concept have been found among some people ..."

4. Relational Model of Daydreaming

Marcel [17] has argued that conscious experience is constructed by a higher-order sensory and cognitive processing. A conscious percept is said to result from the constructive act of fitting a perceptual hypothesis to the lower level data 'to make sense of as much data as possible at the highest and most functionally useful level'. Edelman's [18] ideas are also similarly based on neural processing in the brain. He hypothesizes that consciousness results from stored patterns and current sensory inputs. He identifies the attentional functions with a system composed of the septum, hippocampus and cingulate cortex. This means that consciousness is not of purely cerebral origin and that other areas, especially hippocampus, are also involved.

In recent publications, Taylor [19] has proposed his "relational model of consciousness" following earlier approaches. According to him "consciousness arises due to the active comparison of ongoing brain activity,

stemming from external inputs in the various modalities, with somewhat similar past activity stored in episodic memory". His model is based on a more detailed neurophysiological content and appears to be implementable in comparison with functional models which are meant to explain the action of the brain to achieve conscious processing. Taylor's model is interpreted partly as selective attentional (NRT) and partly as relational(comparison of current stimuli with episodic memory of past events). We first describe the relational model as shown in the block-diagram of Fig 1 , where each block represents a cortical or mid-brain structure. The episodic memory (EM) contains earlier experiences , stored after suitable initial encoding in the preprocessor working memory net (PM/WM). The comparison in D is performed to determine the closeness of the input and the stored episodic memory.

Taylor's model involves the hippocampus, the limbic system, the extra-striate occipital areas and the Nucleus Reticularis Thalami (NRT). The NRT is a wafer of neurons continuing upward between the external medullary lamina and the internal capsule from the zona incerta. Its axons point toward the thalamas .The reticular cells are exclusively inhibitory. The reticular nucleus is uniqely positioned to influence the flow of information between the thalamas and the cortex. According to Taylor the mental content of an experience consist of relations of external inputs of various modalities to stored memories of relevant past experiences : "These past memories need not surface as conscious experience in themselves , but are regarded as constraints or guides to further experience .On the other hand , they may become conscious in their own right , so that one is reminded of past experiences as one lives through the stream of one's own conscious thoughts. These are constantly surfacing into consciousness and then disappearing again, as if continually flowing along just below the surface, only to erupt occasionally when the memory becomes silent. They are also in constant competition with each other, new inputs and their related memories fighting to become new sources of action or of higher cognitive processing such as reasoning or planning". The above statement of Taylor aptly describes the experience of daydreaming. Daydream thoughts may arise directly from memory , or they may be elicited by external stimuli of distracting type or even by a stimuli belonging to the task at hand [9]. Mental images (visual, tactile, auditory) are common in fanciful daydreams.

Taylor argues for three different processors : (a) Relational Processing : Preprocessed coded inputs in working memory(WM) and related episodic memory (EM) for past events, (b) Local Intramodular (WM) Competition : Between nodes coded for different

interpretation of the same input ,for resolving ambiguity, and (c) Global Competition : Between activities on the various working memory systems. Fig 2 shows how daydream thoughts emerge into consciousness. The model shows two PM/WM pairs coupled to an episodic memory (EM) and to the TH-NRT-C system for the global competition. Daydream thoughts are produced from EM based on emotional input generated by some neuro modulator effects.

5. The Role of Recurrent Neural Networks

The feedforward neural networks using backpropagation learning rule are used as a major tool in PDP modeling applications. These have to be driven by the environment. Recurrent neural networks , on the other hand, can build representations internally and are capable of maintaining a level of activity in the absence of external input. As pointed out by Churchland [8] " the recurrent pathways provide access into the networks immediate past .You will now appreciate that a recurrent network's capacity for generating prototypical event sequences gives it a window into the future as well". Explaining later, he gives seven dimensions of consciousness including this one: Consciousness "independent of sensory inputs" : " one can daydream about the future , search through one's memories , or address and pursue a complex problem in one's imagination , all without input from the senses ..."

Churchland presents the recurrent network as a device which can display functional analogs of all seven dimensions of consciousness. A number of other authors have recently proposed the importance of recurrent neural networks in connectionist models as "active symbols"[20], recognition codes"[21],and "cell assemblies" [13]. In a recent publication O'Brien and Opie [22] have proposed "stable activation patterns" in a connectionist vehicle theory of consciousness. They write " ..the existence of stable patterns of activities at the level of neural networks is quite consistent with seamless nature of our own ongoing phenomenal experience .. such stabilization can occur very rapidly, it is possible for the neural network to generate many stable states per second ..."

The fundamental building block of a connectionist model for representing concepts as "active symbols" is based on recurrent networks which are capable of supporting activity over time. These networks can represent internally persisting activity in the absence of external percepts. In the relational model described in section 4 , each WM can be modeled by a recurrent network.

6. The Role of Emotion in Daydreaming

The affective disposition controls an individual's daydreams. The limbic system is the seat of emotions[23].Specific regions in the limbic system evaluate sensory inputs in terms of good-bad, pleasant-painful, rewarding-punishing, satisfactory-aversive. The ability to discriminate good from bad and to evaluate whether an activity is rewarding or punishing is critical to the selection and control of behavior by an agent. The limbic system provides this capacity for evaluation to a daydreaming robot. The generation of daydream scenarios are controlled by emotion. If the evaluation by the limbic system is negative the robot will refrain from enacting what it planned to do. Such evaluations are also useful in controlling the storage of events.

Episodic memory is autobiographical memory for a person's past experiences , thoughts and actions. It is part of the self-concept. It's role in almost all aspect of cognition is well-recognized. Daydreams are generated when emotion influences the episodic memory Fig 2. In the DAYDREAMER program [3] an intuitive set of seven control goals, activated by emotions, trigger and direct daydreaming. The representation of emotion in DAYDREAMER uses low-level signals with magnitude and sign and other optional parameters. Recently some effort has gone into constructing connectionist models for emotion [23]. Some models use two interacting neural networks approximating the functions of the limbic system and the neocortex , both networks influencing each other. This is a significant step of combining emotions with memory and can be employed in the connectionist model for daydreaming

7. Conclusion

This paper has presented a framework for a connectionist model of daydreaming in robots. We have proposed a modular block diagram, based on a relational model of consciousness to show how daydreams can be generated by an interplay of imagination, emotion and memory. As a next step we will look for detailed neural networks for each module and provide simulation results. It is hoped that some of the ideas can be experimentally verified by simulating a "society of daydreaming software robots" living in a virtual environment and communicating through facial expression, gestures and movements instead of using the English language interfaces as used by the DAYDREAMER program.

References

[1] James W (1890-1983),The principle of psychology. Dover, New York.

[2] Dennet DC (1996), Kinds of minds: toward an understanding of consciousness. Basic Books, New York, pp. 11-15.

[3] Mueller ET (1990),Daydreaming in humans and machines - A computer model of stream of thought. Ablex , New Jersey.

[4] McCullock WS, Pitts WH (1943), A logical calculus of ideas immanent in nervous activity. Bull.Math.Biophysics., New York, pp. 115-133.

[5] Baars BJ (1988), A cognitive theory of consciousness, Cambridge University Press.

[6] Dennet DC (1991), Consciousness explained. Allen and Lane, New York.

[7] Taylor JG (1999), The race for consciousness, MIT Press, Cambridge, MA.

[8] Churchland PM (1995), The engine of reason , the seat of the soul : A philosophical journey into the brain. MIT Press, Cambridge, MA.

[9] Farthing GW (1992), The psychology of consciousness, Prentice Hall, New York.

[10] Klinger E (1978), Modes of normal conscious flow, In [11], pp. 225-258.

[11] Pope KS, Singer JL (ed), (1978) The stream of consciousness. Plenum Press, New York and London.

[12] Brooks RA (1986), A robust layered control system for mobile robots. IEEE Journal of Robotics and Automation 2(1): pp14-23

[13] Braitenberg V (1984),Vehicles : experiments in synthetic psychology, MIT Press, Cambridge, MA.

[14] Papert S (1993), The children's machine. Basic Books, New York.

[15] McCarthy J (1995), Making robots conscious of their mental states. Computer Science Department, Stanford University.

[16] Sloman A (1999), Architectural requirements for human-like agents both natural and artificial. School of Computer Science, University of Birmingham(U.K.).

[17] Marcel AJ (1983), Conscious and unconscious perception : Experiments on visual masking and word recognition. Cognitive Psychology 15: 197-237

[18] Edelman GM and Mountcastle VB (1982), The mindful brain. MIT Press, Cambridge, MA.

[19] Taylor JG (1998), Constructing the relational mind. PSYCHE 4(10):

[20] Kaplan, S. Wearer M, French RM (1992), Active symbols and internal models: Towards a cognitive connectionism. In : Clark A, Lutz R (ed), Connectionism in Context, Springer-Verlag, London, 1992, pp.91-110.

[21] Grossberg S (1987), Competitive learning : from interactive activation to adaptive resonance. Cognitive Sci. 11 : 23-63.

[22] O'Brien G, Opie J (1999), A connectionist theory of phenomenal experience. Behavioral and Brain Sciences, 22 (1):

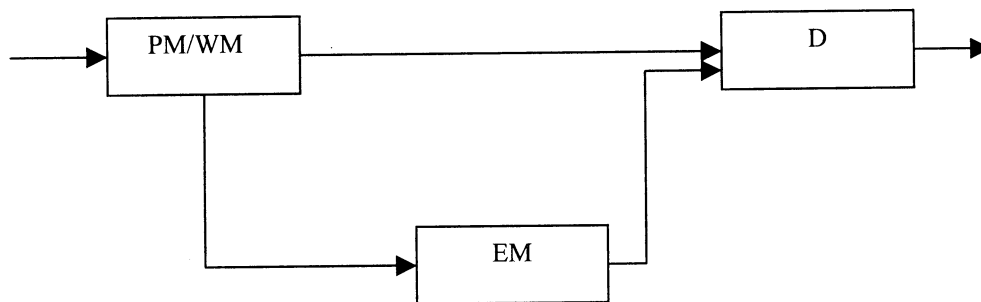[23] Picard R (1997), Affective Computing. MIT Press, Cambridge, MA.
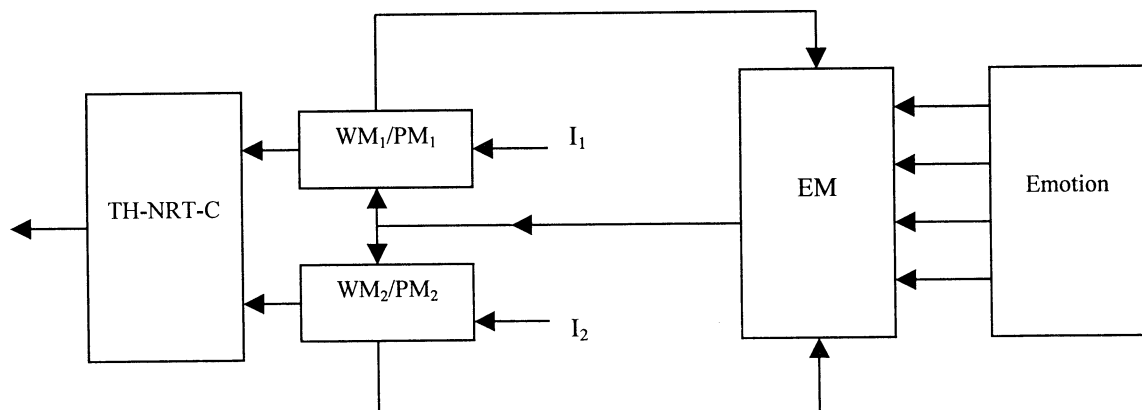
Fig.1. The Relational Model

Fig.1. Local & Global Competition in Daydreaming

Four Layered Robot Control Architecture

J.J. Collins, Lucia Sheehan, Malachy Eaton and Michael O'Neill
Dept.of Computer Science and Information Systems
University of Limerick,
Limerick. Ireland.
Email: [j.j.collins, lucia.sheehan, malachy.eaton, michael.oneill]@ul.ie

Abstract

We describe an extension to the traditional three layer robot control architecture to incorporate a fourth cognitive layer. It is proposed that this cognitive layer enables abstract learning at the conceptual level. In this context we adopt the problem of pursuit and evasion as the domain in which to benchmark the system, subscribing to the view that the holistic decomposition of the control architecture will result in an improvement in efficiency of both learning and navigation, and from a system designers perspective to specify and monitor the interface between these two.

1 Introduction

Within the field of robot control, a system design based on a three-layer architecture has emerged as the de facto standard [8] as shown in fig.1. This consists of:

- A computationally expensive off line deliberative or planning layer at the top level.

- A sequencing layer which coordinates communication and control between the other two layers.

- An online reactive control layer with minimal internal state and tightly integrated sensory motor pathways.

The three layer architecture largely ignores issues such as sensor processing, learning and world modelling. Described here is recent work which extends the three layer architecture, see fig.1, to incorporate a fourth cognitive layer as depicted in fig.3, grounded within an application framework based on robot pursuit and evasion [4]. The fourth layer is based upon a connectionist model using the reinforcement learning paradigm. It is proposed that this extra layer enables abstract learning at the conceptual level.
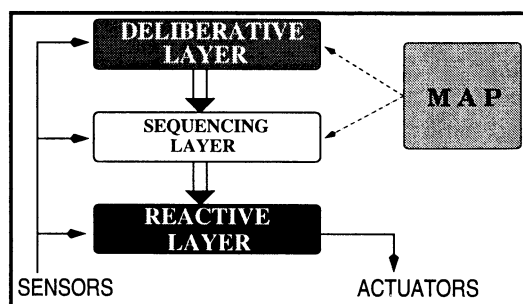
Figure 1: Generic three-layer robot control architecture.

2 Robot Pursuit and Evasion

The current trend in standard benchmarks for evaluating control architectures is in competitions such as RoboCup. It is difficult, however, to differentiate actions arising out of such complex interactions and concurrent goals in a multi-agent system. We subscribe to the view that robot pursuit and evasion represents a much more feasible and theoretically grounded domain in which to benchmark architectures.

Interaction amongst animat agents is being studied within the context of pursuit and evasion. This is based on the derivation of pursuit-evasion behavioural strategies in autonomous agents with a reinforcement learning paradigm as the tactical engine of the system architecture. Pursuit-evasion strategies have strong ethological, biological and game-theoretic foundations [3], and constitute 2nd order level of complexity in autonomous agent research, thus requiring highly adaptive and robust behaviours to ensure survival. This may also yield insight into integration of sensory-motor loop, and social psychology or Machiavellian intelligence, and provide an impetus to a resolution of the cognitive versus behaviour debate. Current focus is on strategies learned by the pursuer to predict and counteract the evader's tactics of (A) long term deliberative directional avoidance, and (B) short term
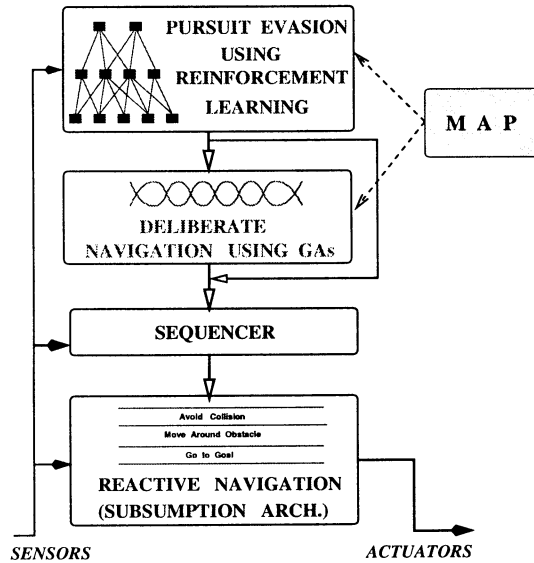
Figure 2: Proposed four layer robot control architecture for pursuit and evasion. The third and top layer in traditional designs has been extended with a game theoretic cognitive layer.

reactive protean behaviours typified by erratic zigzagging.

3 Four Layered Architecture

To resolve some of the issues mentioned previously, the generic three layered architecture has been augmented with one additional layer.

3.1 The Reactive Layer

The reactive layer is a subsumption architecture as shown in fig.3, designed using seven behaviours to implement an online goal finding capability. The behaviours are based on fuzzy logic and specified using a behaviour grammar as provided in the Saphira [1] simulator and environment, and embodied by Pioneer [2] robots. This reactive layer maintains the integrity of the robot at all times and is interfaced with the navigation layer to implement more efficient navigation in partially unknown environments.

3.2 The Sequencing Layer

The sequencing layer is provided by the Saphira architecture itself and is based upon a Procedural Rea-

---

[1] Saphira ©is provided by SRI International
[2] Pioneer ©is a trademark of ActivMedia
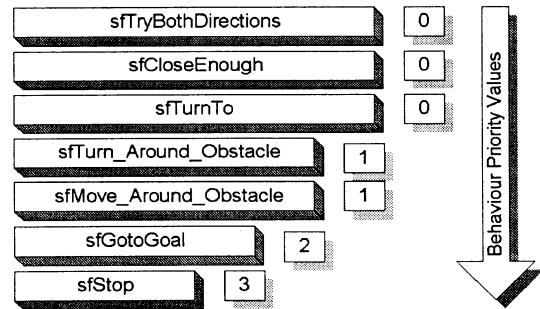
Figure 3: Schema of the subsumption architecture which consists of seven concurrently executing behaviours.

soning System (PRS)[15].

3.3 The Deliberative Navigation Layer

The navigation layer uses a genetic algorithm (GA) to evolve the path that the robot must travel to reach its target. The GA uses a variable length genotype and an enhanced operator set such as insert and delete. The GA evolves a path to traverse a specific topography which is currently given a priori. The addition of a cognitive layer speeds up learning by reducing the dimensionality of the learning problem and decomposing it into two independent tasks. The top connectionist layer is a domain independent, game theoretic, cognitive module that learns strategies for pursuit or evasion.

3.4 The Fourth Layer

The role of the fourth layer is to enable abstract learning to facilitate a decomposition of the higher level learning from the deliberative navigation. Potential benefits arising from this are an improvement in efficiency of both learning and navigation, and from a system designers perspective to specify and monitor the interface between these two. Currently this is implemented using a variant of temporal difference lambda learning, known as Sarsa Lambda[17].

4 Navigation

Efficient navigation is critical for autonomous mobile robots [14] and constitutes a basis upon which strategic and tactical adaptive complex behaviours may be built such as pursuit-evasion. The mobile robot problem is as follows: given a description of a known environment, plan a path from a start to goal position

which avoids collision and satisfies other user specified constraints [22].

Characteristics of the navigation problem is that it is PSPACE-hard [16], and in general the computational cost of path planning increases exponentially [12]. Navigation can be decomposed into three fields:

- Off line planners - compute a path prior to commencement of travel. World must be completely specified, and thus environment is static, see [12] for surveys.

- On line planners - using heuristics embedded in a behaviour oriented subsumption architecture, respond to stimuli in the environment as the robot travels in the direction of the goal [1, 13, 19]. The environment is not specified and thus can be dynamic.

- Hybrid: combination of off line and on line planning for both static and dynamic environments [5, 6, 7, 22].

Off line planners assume that the world is totally known and try to find a shortest path route as a solution to the navigation problem. Traditional optimization techniques have proved costly for this problem type.

Online planners are generally purely reactive stimulus response systems, and the level of path optimization is directly proportional to the efficiency of the implementation. Reactive architectures suffer from two shortcomings [14]: a) they may generate highly inefficient trajectories since the next action selected is a response to the current situation only b) difficult to program. The use of learning paradigms within a connectionist network architecture to derive online navigation [14] has yielded more fruitful results than purely reactive subsumption architectures. However, the limitations of reactive systems are now replaced by the problems of design of the network topology, choosing a suitable training set, network convergence during training and generalisation to unknown environments.

Hybrid approaches have relied on reactive navigation while incremental map building takes place concurrently. When the map is built, planning then takes over from reaction as the driving mechanism. A criticism of this approach is that building and maintaining maps is far from trivial, since noisy sensory input and dead reckoning errors will exponentially propagate errors into the mapping process. Unless localisation and position tracking are implemented [11, 21] to resolve introduced errors and uncertainty, the derived maps will invalidate the control architecture.

Genetic Algorithms (GAs) have proved to be very effective in search and optimization problems which are not amenable to traditional methods [10]. We adopt the GA methodology for the task of navigation based upon the work of Xiao et al.[22]. The advantage of this approach is that we can obtain near-optimal solutions in an efficient manner. Currently a world map must be supplied a priori. The standard set of genetic operators were extended with a set of new operators such as insert, delete, prune, and smooth specific to this problem domain. A self adaptive tuning mechanism was adopted to optimise the probabilities associated with these operators.

5 Discussion

To date, the reactive and deliberative layers have been designed and implemented. Current focus is on the cognitive fourth layer. Amongst the many issues to be resolved are the correspondence problem - aligning the agent's internal model of the environment with input from noisy sensor channels, and benchmarking of the soft controllers.

Biological entities are thought to exist in and adapt to dynamic situations through the use of a diploid genotype. Each chromosome consists of two genetic strings which are combined in the genotype to phenotype mapping. The second string allows previous solutions to be held in abeyance until the environment changes, and the solutions are required again. In biological entities, knowledge is stored in the genetic material or genotype, which constitutes genetic memory. Learning is achieved through gradual adaptation of the genotype, the learned knowledge being made available to the subsequent generation - phylogenetic learning [20].

We propose to evaluate diploidy as a representation for navigation in dynamic environments, in which previously unknown objects in the world are detected. Based on the initial sensor readings, and the view angle of the readings, heuristics will be deployed to estimate the topology of the object. Using these heuristics, the diploid GA will continue evolving a new path around the object. Diploidy has been shown to react much faster to changes in dynamic environments than a haploid genotype as used by Xiao [10]. Empirical studies have shown that the use of a genotype based on polygenic inheritance will facilitate much faster adaptation to changes in dynamic environments [18]. Thus, our immediate goal is to evaluate diploidy versus polygenic inheritance for navigation in dynamic environment.

References

[1] Brooks, R. (1986). A Robust Layered Control System for A Mobile Robot. *IEEE Trans. on Robot Automation*, 2:14–23.

[2] Brooks, R. (1991). Intelligence without Representation. *Artificial Intelligence*, 17:139–159.

[3] Cliff, D. and G. F. Miller (1996). Co-evolution of pursuit and evasion II: Simulation methods and results. In: *Fourth Int. Conf. on the Simulation of Adaptive Behaviour (SAB'96)*. pp. 506–515.

[4] Collins, J.J., L. Sheehan and C. Casey (1998). Evolutionary path generation for navigation in mobile robots. To appear in: *7th European Workshop on Robot Learing (EWLR98)*, July 1998, Edinburgh.

[5] Connell, J. H. (1992). SSS: A Hybrid Architecture Applied to Robot navigation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 2719–2724.

[6] Foux, G., Heymann, N. and Bruckstein, A. (1993). A Robust Layered Control System for A Mobile Robot. *IEEE Trans. on Robot Automation*, 9:96–102.

[7] Gat, E. (1991). Integrating Reaction and Planning in a Heterogeneous Asynchronous Architecture for Mobile Robot navigation. *SIGART Bulletin*, Vol. 2, NO. 4, pages 70–74.

[8] Gat, E. (1998). Three Layer Architectures. In Kortenkamp, Bonasso and Murphy, editors *Artificial Intelligence and Mobile Robots*, pages 195–210. AAAI/MIT Press.

[9] Georgeff, M. P., and Ingrand, F. F. (1989). Decision-Making in an Embedded Reasoning System. In *Proc. of the AAAI Nat. Conf. on Artificial Intelligence*, pages 972–978.

[10] Goldberg, D. (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing.

[11] Koenig, S, and Simmons, G. (1998). Xavier: A Robot Navigation Architecture Based on Partially Observable Markov Decision Process Models. In *Artificial Intelligence and Mobile Robots*, pages 91–122. AAAI/MIT Press.

[12] Latombe, J. (1991). Robot Motion Planning. Kluwer Academic Publishers.

[13] Lumelsky, V. J. (1991). A Comparative Study on the Path Length Performance of Maze-searching and Robot Motion Planning Algorithms. *IEEE Trans. on Robot Automation*, Vol. 7, No. 1, pages 57–66.

[14] Millán, J. (1996). Rapid, safe, and Incremental Learning of Navigation Strategies. *IEEE Trans on Man, Systems and Cybernetics*, Vol. 26, No. 3, pages 408–420.

[15] Myers, K.L. (1996). A procedural knowledge approach to task level control. In *Proceedings of the third international conference on AI planing systems*. AAAI Press.

[16] Reif, J. H. (1979). Complexity of the Mover's Problem and generalizations. In *Proc. of the IEEE Sym. on Foundations of Computer Science*, pages 421–427.

[17] Rummery, G.A., Niranjan, M. (1994). On-line Q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166. Engineering Department, Cambridge University.

[18] Ryan, C. and Collins, J.J. (1998). Shades: A polygenic inheritance scheme that outperforms diploidy. To appear in: *Parallel Problem Solving from Nature (PPSN'98)*, Sept. 1998, Amsterdam.

[19] Schoppers, M. J. (1987). Universal Plans for Reactive Robots in Unpredictable Environments. In *Proc. 10th Int. Joint Conf. on Artificial Intelligence*, pages 1039–1046.

[20] Sipper, M., Sanchez, E., Mange, D., Tomassini, M, Pérez-Uribe, A. and Stauffer, A. (1997). A Phylogenetic, Ontogenetic, and Epigenetic View of Bio-Inspired Hardware Systems. *IEEE Trans. on Evolutionary Computation*, Vol. 1, No. 1, pages 83–97.

[21] Thrun, S., Bücken, A., Burgard, W., Fox, D., Fröhlinghaus, T., Hennig, D., Hofmann, T., Krell, M. and Schmidt, T. (1998). Map Learning and High Speed Navigation in RHINO. In Kortenkamp, Bonasso and Murphy, editors *Artificial Intelligence and Mobile Robots*, pages 21–52. AAAI/MIT Press.

[22] Xiao, J., Michalewicz, Z., Zhang, L., and Trojanowski, K. (1997). Adaptive Evolutionary Planner/Navigator for Mobile Robots. *IEEE Trans. on Evolutionary Computation*, Vol. 1, No.1 pages 18–28.

Automatic Generation of Robot Behaviours using Grammatical Evolution

Michael O'Neill, J.J. Collins & Conor Ryan
Dept. Of Computer Science And Information Systems
University of Limerick
Limerick, Ireland
Email: [Michael.ONeill,J.J.Collins,Conor.Ryan]@ul.ie

Abstract

We describe an evolutionary approach to the automatic generation of behaviours for a subsumption architecture. The evolutionary algorithm, called Grammatical Evolution, can output code in any language. The output language is described in the form of a Backus Naur Form grammar definition, and as a result in this context employs a behaviour grammar. We begin with a description of the Grammatical Evolution algorithm, and then demonstrate the ease with which behaviours may be generated using this approach.

1 Introduction

This paper describes an approach to the automatic generation of high level robot behaviours as part of a subsumption architecture using Grammatical Evolution (GE). GE is an evolutionary algorithm which can automatically generate a computer program in any language from a problem description. It is distinguished from Genetic Programming (GP) by it's genotype to phenotype mapping, that is, the population contains variable length binary strings which must undergo a mapping process in order to generate a program, unlike the direct representation of a program in the form of a parse tree in GP. As the mapping process uses a BNF grammar definition to describe the language of the output code, and this definition is a plug-in component, GE can be used to generate code in any language. Another feature of GE's mapping process is its degenerate genetic code, which allows a more efficient evolutionary search to be performed by taking advantage of neutral mutation events and the subsequent neutral networks arising from these events. GE has been applied successfully to various problem domains producing algorithms which in some cases are better than those designed by human programmers and also GP.

The aim of this work is to firstly show that GE
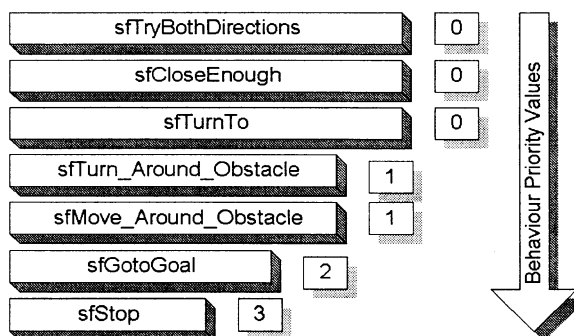
Figure 1: Example of a subsumption architecture which has been currently implemented

can be used with Saphira to evolve a behaviour, before tackling more complex behaviours which interact with and call upon a primitive set of behaviours, and to provide a form of global memory which is potentially accessible by all behaviours. The use of global memory structures when hand coding an architecture has previously been shown to be a non trivial task. To this end the evolving behaviours are plugged into a subsumption architecture (Figure 1) running on Saphira which can then tested on a real robot platform, such as Pioneer or Khepera. One of the hypotheses currently being explored is that relatively complex behavioural practices can be built bottom-up from a set of simple responses. This hypothesis has already been put to the test by Arkin [1] in *Autonomous Robot Architecture* and by Konolige et al. [3]. The compositional approach to building complex behaviour has been given in a formal framework of multi-valued logics by Saffioti, Konolige and Ruspini. The key point is that complex behaviour can demonstrably emerge from simultaneous production of similar responses. However, in practice for a subsumption architecture a limit of 11 to 13 layers exists in terms of complexity for the designers perspective. An example of a hand coded architecture can be seen in Fig. 1.

Figure 2: The Grammatical Evolution System and a Biological System

We describe GE, and show how this system can easily generate behaviours for Saphira using the systems Behaviour Grammar as the grammar definition used in GEs mapping process.

2 Grammatical Evolution

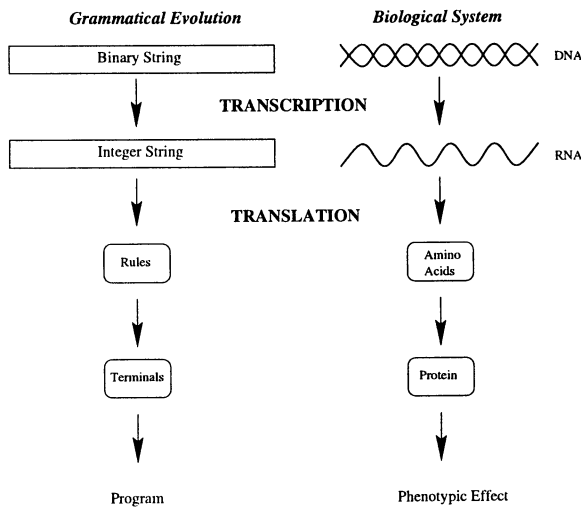Grammatical Evolution (GE) is an evolutionary algorithm which can evolve computer programs in any language. Rather than representing the programs as parse trees, as in GP [5], we use a linear genome representation. Each individual, a variable length binary string, contains in its codons (groups of 8 bits) the information to select production rules from a Backus Naur Form (BNF) grammar, see Fig. 2.

BNF is a notation which represents a language in the form of production rules. It is comprised of a set of non-terminals which can be mapped to elements of the set of terminals, according to the production rules. An example excerpt from a BNF grammar is given below. These productions state that S can be replaced with either one of the non-terminals $expr$, $if - stmt$, or $loop$.

```
S ::= expr      (0)
    | if-stmt   (1)
    | loop      (2)
```

In order to select a rule in GE, the next codon value on the genome is generated and placed in the following formula:

$$Rule = Codon\ Integer\ Value\ \mathbf{MOD}$$
$$Number\ of\ Rules\ for\ this\ non - terminal$$

If the next codon integer value was 4, given that we have 3 rules to select from as in the above example, we get 4 MOD 3 = 1. S will therefore be replaced with the non-terminal $if - stmt$.

Beginning from the the left hand side of the genome codon integer values are generated and used to select rules from the BNF grammar, until one of the following situations arise: (a) A complete program is generated. This occurs when all the non-terminals in the expression being mapped are transformed into elements from the terminal set of the BNF grammar. (b) The end of the genome is reached, in which case the *wrapping* operator is invoked. This results in the return of the genome reading frame to the left hand side of the genome once again. The reading of codons will then continue unless an upper threshold representing the maximum number of wrapping events has occurred during this individuals mapping process. (c) In the event that a threshold on the number of wrapping events has occurred and the individual is still incompletely mapped, the mapping process is halted, and the individual assigned the lowest possible fitness value.

GE uses a steady state replacement mechanism [9], and the standard genetic operators of mutation (point), and crossover (one point). It also employs a duplication operator, which duplicates a random number of codons and inserts these into the penultimate codon position on the genome. A full description of GE can be found in [10] [8] [7] [6].

3 Problem Domain & Preliminary Results

The problem being tackled here is to automatically generate a simple behaviour as proof of concept. To this end we adopt the behaviour grammar as described for Saphira [4]. Saphira is a robotics control architecture developed by SRI International, based upon a client server model, which can be used in conjunction with either a simulated or real robot, see Fig. 3. The robot server has the ability to respond to commands issued by Saphira, and to send information packets back to Saphira about the robots state. Saphira utilises fuzzy control rules to implement behaviours, which are in fact low level control programs. Each behaviour in Saphira has associated with it priority and
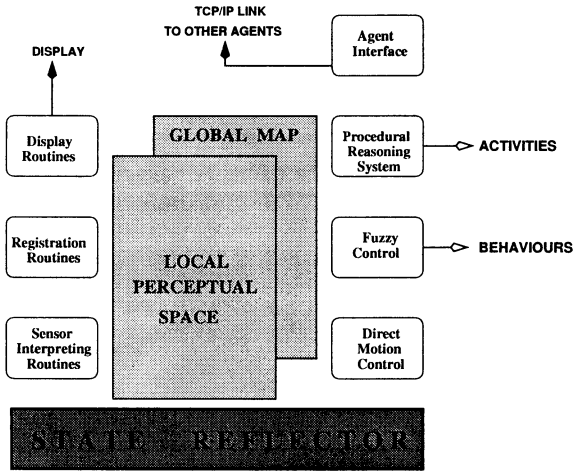
Figure 3: The Saphira control architecture is a set of routines that interpret sensor readings relative to a geometric world model, and a set of action routines that map robot states to control actions. Registration routines link the robot's local sensor readings to its map of the world, and the Procedural Reasoning System sequences actions to achieve specific goals. The agent interface links the robot to other agents in the Open Agent Architecture.

activity values, which determine how it interacts with other behaviours which exist concurrently, and each behaviours routine for invocation.

The actual grammar used by GE is given below.

```
<BEHAVIOUR> :=
            BeginBehaviour ge
            Params
            Rules <RULE_STMTS>
            Update
            Activity <ACT_STMTS>
            EndBehaviour

<RULE_STMTS> :=
If <FUZZY_EXP> Then <CONTROL> <RULE_STMTS>

<FUZZY_EXP> := <FLOAT>
            | Not <FUZZY_EXP>
            | <FUZZY_EXP> And <FUZZY_EXP>
            | <FUZZY_EXP> Or <FUZZY_EXP>
            | ( <FUZZY_EXP> )

<CONTROL> := Turn Left <MOD>
            | Turn Right <MOD>
            | Speed <MVAL>
```

```
<MOD> := <CONST>

<MVAL> := <CONST>

<ACT_STMTS> :=
            Turn <FUZZY_EXP><ACT_STMTS>
            | Speed <FUZZY_EXP><ACT_STMTS>
            | Stop

<FLOAT> := straight_up(<VAR>,<VAR>,<VAR>)
            | down_straight(<VAR>,<VAR>,<VAR>)
            | f_greater(<VAR>,<VAR>,<VAR>)
            | f_smaller(<VAR>,<VAR>,<VAR>)
            | f_eq(<VAR>,<VAR>,<VAR>)
            | <CONST>

<VAR> := sfTargetVel()
            | sfTargetHead()
            | <CONST>

<CONST> := 0.0 | 0.1 | 0.2 | 0.3
            | 0.4 | 0.5 | 0.6 | 0.7
            | 0.8 | 0.9 | 1.0
```

The start symbol of the grammar is $< BEHAVIOUR >$ which specifies the name of the behaviour as being *ge*. In this case the behaviour has no parameters (Params) or Update specified, however, the statements for Rules and Activity will be evolved. The rule statements are comprised of fuzzy rules with each rule recommending only a single action. Saphira carries out the necessary handling for the conjunction of many rules. The activity section of a behaviour describes how this evolved behaviour operates in the larger context of other behaviours in the subsumption architecture. An example behaviour output from this grammar is as follows:

```
BeginBehavior sfStopIt   /* behavior name */
        Params           /* none */
        Rules            /* just one rule */
            If 1.0 Then Speed 0.0
        Update
        Activity
            Speed 1.0
EndBehavior
```

The above behaviour is called sfStopIt, has no parameters or update rules. The Rules section specifies that when this behaviour is used it will always ("If 1.0") set the speed of the motors to 0, i.e. the robot will come to a stop. The Activity section specifies,

Speed 1.0, which means that this behaviour will compete fully with any other behaviours of the same priority for control of the robots speed.

The output from GE for every individual in a population is thus a behaviour in the syntax described by the behaviour grammar, in order to test the behaviour to obtain some measure of fitness we must present it in a suitable format to Saphira. To this end the output behaviour is converted to C code using Saphira's **bgram** utility, and subsequently compiled to a shared object file which can be loaded directly into the executing Saphira client.

The objective of the first experiment is to show that we can generate behaviours using this behaviour grammar with GE. A population size of 50 individuals was used, a probability of 0.9 for crossover, and 0.01 for mutation. The desired behaviour was to move as far away as possible from the robots starting position. The evolutionary process was allowed to run for 20 generations, and successfully evolved behaviours which caused the robot to move away from its starting position.

At the time of writing we were still in the process of conducting the next phase in this work, the evolution of more complex behaviours including obstacle avoidance. Results for these experiments shall be reported at a later date.

4   Conclusion

We have demonstrated that it is possible to evolve simple behaviours for a subsumption architecture using the Grammatical Evolution evolutionary algorithm. Our next objectives are to evolve more complex behaviours from sets of primitives, allowing the behaviours to have access to a form global memory, and the testing of these evolved behaviours on the real world platforms of the Pioneer and Khepera robot architectures.

Acknowledgements

References

[1] Arkin R. 1998. Behaviour Based Robotics. MIT Press, 1998.

[2] Collins J.J., Sheehan Lucia, and Casey C. 1998. Evolutionary Path Generation for Navigation in Mobile Robots. In *Proceedings of the European Workshop on Robot Learning (EWLR'98)*, Edinburgh.

[3] Konolige K., Meyers K.L., Ruspini E.H., Saffiotti A. 1997. The Saphira architecture: A design for autonomy. *Journal of Experimental and Theoretical Artificial Intelligence*, 9:215-235.

[4] Konologie, Kurt. Saphira Software Manual, Version 6.1. 1998.

[5] Koza, J. 1992. *Genetic Programming*. MIT Press.

[6] O'Neill M., Ryan C. 1999. Genetic Code Degeneracy: Implications for Grammatical Evolution and Beyond. In *Proceedings of the Fifth European Conference on Artificial Life*.

[7] O'Neill M., Ryan C. 1999. Under the Hood of Grammatical Evolution. In *Proceedings of the Genetic & Evolutionary Computation Conference 1999*, pages 1143-1148.

[8] O'Neill M., Ryan C. 1999. Evolving Multi-Line Compilable C Code. In *Proceedings of the Second European Workshop on Genetic Programming*, pages 83-92.

[9] Ryan C., O'Neill M. Grammatical Evolution: A Steady State Approach. In *Late Breaking Papers, Genetic Programming 1998*, pages 180-185.

[10] Ryan C., Collins J.J., O'Neill M. 1998. Grammatical Evolution: Evolving Programs for an Arbitrary Language. *Lecture Notes in Computer Science 1391, Proceedings of the First European Workshop on Genetic Programming*, pages 83-95. Springer-Verlag.

Behavior Control of Autonomous Mobile Robots using the Block-based Neural Network

Sang-Woo Moon* and Seong-Gon Kong

Intelligent Signal Processing Lab.
Department of Electrical Engineering, Soongsil University
Seoul, 156-743 Korea
Tel: +82-2-820-0648, Fax: +82-2-816-3639, http://ee.ssu.ac.kr/~skong
E-mail : kagami@ee.ssu.ac.kr, skong@ee.ssu.ac.kr

Abstract

This paper presents a behavior control of a autonomous mobile robot based on a block-based neural network(BNN). The BNN is a novel evolvable neural network model which consists of two dimensional array of basic blocks. Each block has four 'variable' input and output nodes in order to be able to have flexibility in structure. The structure of a BNN is suitable for digital hardware implementation using reconfigurable hardware such as field programmable logic arrays, since internal architecture changes in according to configuration binary strings. Internal structure and connection weights are optimized autonomously and adaptively in real time according to evolution of configuration binary strings using the evolutionary algorithms. Through extensive simulations, the controller designed using a BNN successfully show intelligent navigation and obstacle avoidance behaviors of autonomous mobile robots with limited capability.

Keywords : hardware evolution, evolvable neural network. autonomous mobile robot, evolutionary algorithms, reconfigurable hardware.

1. Introduction

Neural network model that is based on human brain is a good solution for solving some problems that is more complicated with existing other methods. But determination of networks structure and local minima problem are weak points of neural networks. This is why evolutionary algorithm is used for solve this problem..[1][2] In addition, current neural network model is not easy to implement for digital hardware.

Evolvable hardware means specific hardware that can adapt many different surroundings by changing internal logic itself. And it has a characteristic of robustness for noise from inside or outside. The major part of a neural network model uses sigmoid type activation function and depend on connections among the neurons. This is why existing neural network model has been implemented hardware with analog part, not entirely digital!

This paper is proposed Block-based Neural Network(BNN). BNN is consisted of two-dimensional array of basic blocks.[3] A different composition of basic blocks is changed a structure of BNN. Integer weight with limited range and symmetric saturating linear activation function is used. Because of optimization using evolution procedure with binary string that explains network structure and weights, BNN can be easily implemented by digital reconfigurable hardware such as FPGA that can be reconfigurable by configuration bit

string.

This paper explain a BNN model, its optimization method, and experiment on autonomous mobile robot with BNN as a controller for navigation and obstacle avoidance. We obtain a optimal BNN and confirm a performance of navigation and obstacle avoidance through a simulation.

2. Block-based Neural Network Model

2.1 Architecture

The BNN is consisted of two-dimensional array of blocks. Fig. 1 is presented a structure of 3×4 BNN based on basic blocks. Fist and last row of BNN is input and output stage, respectively. And the BNN can contain more that one middle stage. Any block in a middle stage is connected with all four neighbor blocks and a arrow between each blocks is presented a data flow. A left and right most block is connected with each other.
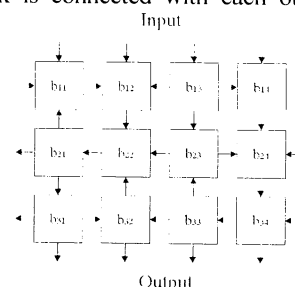
Fig. 1: Structure of the block-based neural network

Each block b_{ij} is a simple feedforward neural network having 4 neurons. The block can have 4 different types depending on input-output connections. All the weights have integer values in order to make hardware implementation easy. In Fig. 2, x_i and y_i indicate input and output respectively. Type A and type B have number of input/output 1/3 and 3/1. Type C and type D are of 2/2 but with different internal connection.

type A   type B   type C   type D
Fig. 2: Internal structure of basic blocks of the BNN

Sigmoid activation function restricts hardware

implementation by exponential term. Therefore, input and output neuron is activated by linear and symmetric saturating linear function, respectively. Normally, symmetric saturating linear have a unit slope. For more continuous output space with restricted training pattern, a slope of activation function should be changed.
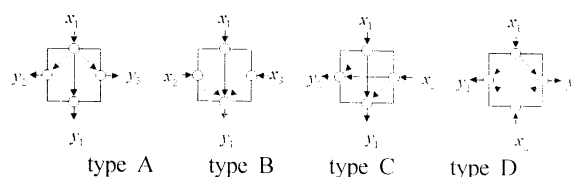
Because the BNN can have various problem-dependent structure mixed with feed forward and backward characteristics, stabilized output resulted from system input is only admitted system's real output. The BNN's feedback characteristics among the block in a same stage or between neighbor stages influence a time for calculation of real output. All data flows of each block determine a neuron's in/out form and internal structure of all blocks.

2.2 Characteristics

The $m \times n$ BNN can be represented successfully any multi layer perceptron(MLP) if number of neuron in each layer n is less or equal than 4. For example, 3bit odd parity pattern can be classified with $3 - 3 - 1$ MLP. The 2×3 BNN can also be classified successfully. Because all weights of the BNN is restricted to integer values, search space is discontinuous. Therefore, in the case of lack of training data such as XOR pattern, learning with evolutionary algorithm have a tendency that a output for the BNN is changed on a sudden. BNN has less complexity for structure optimization rather than neural network that can be admitted random connection among each neurons.

3. Genetic Algorithm for Optimization of a Block-based Neural Network

A evolution objects of the BNN can classify with two different point, determination of in/out form and internal weights of all blocks, and is optimized by genetic algorithm. A genetic algorithm have a characteristic of global search and can optimize specific objective using a fitness function for problem-dependent objectives.[4] This paper is redefined encoding, reproduction, and crossover method for optimization of the BNN. Internal copy operator is also used.

3.1 Encoding

A data flow of each block determine each block's in/out form. In/out form of each block determine internal structure of each block. Finally, structure of the BNN is determined a data flow of each block. A data flow of each block and weights should be encoded for evolution simultaneously. Current multi-dimensional encoding assign each gene for each dimension. But, in this paper, a data flow of each block and weights consist of two-dimension chromosome to progress of performance for crossover at once.
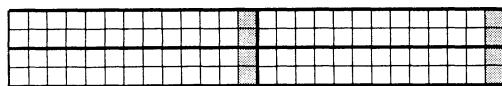
(a) chromosome representation for the BNN
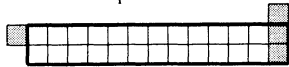
(b) two-dimensional encoding for one block.

Fig. 3: Two-dimensional encoding for the BNN

Fig. 3(a) presents a example of a individual that is composed by four encoded blocks. All weights are encoded by 6bit binary. □ is defined a weight, ▨ is presented data flow of each block and is defined 0 in case of ↓ and ←, 1 in case of ↑ and →. Fig. 3(b) presents two-dimensional encoding for only one block. structure bits at upper left and upper right corner are shared with neighbor blocks. Upper right structure bit of a block in first stage of the BNN is always set to 0, and lower right structure bit of a block in last stage of the BNN is always set to 0. Weights is expressed from $-(2^{l-1}-1)$ to $2^{l-1}-1$ with two zero that means disconnection.

$$w = w_b - 2^{(l-1)} + s(w_b) \qquad (1)$$

$$s(w_b) = \begin{cases} 1 & \text{if } w_b < 2^{l-1} \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

Eq. (1) is transformed w_b into w. w_b is positive integer and is expressed by l-bit length binary.

3.2 Genetic operators

Reproduction mechanism normally used is a stochastic selection using a roulette wheel. Roulette wheel selection method is based on fitness ratio and have some weak points, mature convergence and genetic drift. Linear scaling method can deal with this two problems. Linear scaling is a method that scale all fitness using maximum, minimum, and average fitness by linear function. Linear fitness scaling can consider about distribution state of all fitness, but if average fitness is to be close maximum fitness, a fitness less than average can be evaluated to negative value. To prevent negative fitness evaluation in linear scaling procedure, scaling procedure parameter based on linear function is changed from maximum and average fitness to average and minimum fitness at condition for negative fitness evaluation. In this case, after moderate convergence, genetic drift is occurred. To overcome the weak point of linear fitness scaling, modified linear scaling method is used. Modified fitness scaling method use two different linear function based on maximum, minimum, and average fitness for avoidance of negative scaled fitness and genetic drift.

- Case 1 : $f_{\min} \geq f_s$

$$f'(x_i) = a_1 x_i + b_1 \qquad (3)$$

- Case 2 : $f_{\min} < f_s$

$$f'(x_i) = \begin{cases} f'_1(x_i) = a_2 x_i + b_2 & \text{if } x_i < f_{avg} \\ f'_2(x_i) = a_2 x_i + b_2 & \text{otherwise} \end{cases} \qquad (4)$$

Eq. (3) and (4) is linear scaling functions at each cases, and parameters in scaling functions are

$a_1 = \frac{(\alpha - 1)f_{avg}}{f_{max} - f_{avg}}$, $b_1 = \frac{f_{avg}(f_{max} - \alpha f_{avg})}{f_{max} - f_{avg}}$, $a_2 = \frac{f_{avg}}{f_{max} - f_{avg}}$, $b_2 = \frac{-f_{min}f_{avg}}{f_{max} - f_{avg}}$.

$f(x_i)$ is a fitness of i-th individual, and $f'(x_i)$ is a fitness after scaled process. $f_{\min}$, f_{avg}, $f_{\max}$ mean minimum, average, and maximum fitness by object function, respectively. Also, in above eqs., f_s is used as $(\alpha f_{avg} - f_{\max})/(\alpha - 1)$ for criterion value to confirm the case of $f_{\min} < 0$ after procedure of normal linear fitness scaling. α is a constant of linear scaling and effects a slope of linear scaling function. Together with modified linear fitness scaling method, additional methods for selection, elitist, is used.

Genetic operator should be also modified to be compatible with two-dimensional encoding. A individual composed by multi-dimensional encoding can consider the three extended types of crossover operation compared with a case of one-dimensional encoding. At first, there is a method to select only one crossover point for each dimension. Next, there is a method to select multi crossover point for each dimension. Finally, and variable crossover point for each dimension is permitted with the condition of fixed total number of crossover point $\sum_{i=1}^{n} k_i = k$ ($k_i \geq 0$). Final method is generalized form rather than the others. Through the generalized form of multi-dimensional crossover, more various schema can be created rather than the other two multi-dimensional crossover operation.
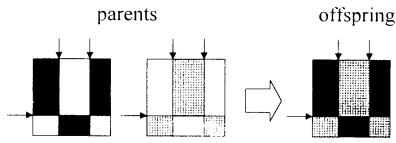

Fig. 4: Example of operation for generalized 2D crossover

Fig. 4 is a example of generalized multi-dimensional crossover operation in the case of $k_1 = 1$, $k_2 = 2$. In the case of generalized two-dimensional crossover, like a one-dimensional crossover operation, crossover operation with one and two crossover point results in maintaining a individual's variety and improving a convergence, respectively.

Two different probability of mutation is applied to a structure and weights bit in binary representation. p_{mc} is a probability of mutation for structure part, and p_{mw} is also a probability of mutation for weights part.

A transmission of data between blocks have in forward and backward characteristics. Therefore, to improve evolution performance, both two characteristics are considered. Internal copy operator play an important role for an improvement of evolution performance of neural network included a feedback element[2]. In this paper, Internal copy is used to help crossover for sufficient variety of all individuals. In populations expressed by two-dimension encoding, individuals selected by internal copy probability p_{ic} perform a copy operation separately from some part of a chromosome to other part of itself. A size and location for copy in a selected chromosome is determined randomly.

4. Robot Control with Block-based Neural Network

BNN is designed for realization of digital hardware and it is more suitable for a part of discrete signal such as control field, not normal signal processing. In this paper, navigation and obstacle avoidance is a objective of a robot control.[5] And BNN is used as a controller of robot. We show a control of robot by optimized BNN through a evolution procedure that have a specific objective in fitness function.

4.1 Robot and Environment Modeling

For offline simulation, robot modeling in software simulator is based on Khepera robot.(made by LAMI) A robot model is considered with location of sensors, sensing range, size of a robot, width and distance of a navigation path, and etc. At modeling a robot, we don't use a backward sensor because robot is assumed only a forward movement and rotation on current place. And only one forward sensor is used for robot model, not two. To reduce a gap between simulation and real experiment, we assume a sensing range to be similar with real khepera robot. A maximum distance of sensing is a 32mm, and diameter of a robot is assumed to be 48mm.

A robot can only use a data from its sensor for recognition of surroundings. A sensor that is equiped for real khepera has relatively very short sensing range. Therefore, if a width of a path is too much wide to recognize a obstacle, a robot loses a directional characteristic in its navigation. In a simulation environment, a robot should know a location for end of simulation. A path used in a simulation is composed with 128mm width path.

4.2 Block-based Neural Network as a Robot Controller

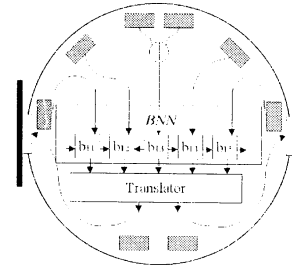1×5 BNN is used for controller of robot.

Fig 5: BNN for robot control

Fig. 5 shows a BNN as a robot controller that has a five input from proximity sensor and output for translator which change a BNN's output to motor action for robot. BNN's output is interpreted to a rotation angle in current position.

Table 1: Analyzation of output patterns for BNN

y_4	y_3	y_2	y_1	Rotating Angle
-1	-1	-1	1	$+45°$
-1	-1	1	-1	$-45°$
-1	+1	-1	-1	$+90°$
1	-1	-1	-1	$-90°$
else				$0°$

Table 1 presents a relation between BNN's output and rotating angle. A repeated rotation and forward movement are a basic navigation mechanism of a modeling robot.

4.3 Fitness Function

Genetic algorithm has a good point for object dependent optimization by definition of fitness. In this paper, we define a objective of robot control as a navigation and obstacle avoidance. There is no meaning about navigation through optimal path because of lack of prior-information about navigation path. We can assume that any location should not pass twice. As a navigation path is not a maze, this assumption is a condition of optimal navigation. In this paper, less collision and high

speed is a check point of fitness evaluation for optimal movement.

$$f(c, p) = (1 - \frac{c}{c_{max}})(1 - \frac{s}{s_{max}})$$ (5)

$$s = \sum_{k=1}^{m-1} \sum_{j=1}^{3} |s_j(k) - s_j(k-1)|$$ (6)

Eq. (5) present a fitness function for robot control. c is a number of collision. s is a change of sensor input $s_j(k)$. m is a number of maximum step for movement of robot. We assume that a robot moves 16mm per step. c_{max} and s_{max} is a maximum number of c and s. In the case of $c > c_{max}$ and $s > s_{max}$, we set a $c = c_{max}$ and $s = s_{max}$, respectively. These cause fitness value by 0.

4.4 Simulation

We assume that a robot with optimized BNN in training path can navigate and avoid obstacle at any other path. In a training path, optimal BNN is obtained by genetic procedure and tested in other navigation path for confirmation of above assumption. For a control of robot, 20 individual is used for genetic algorithm.
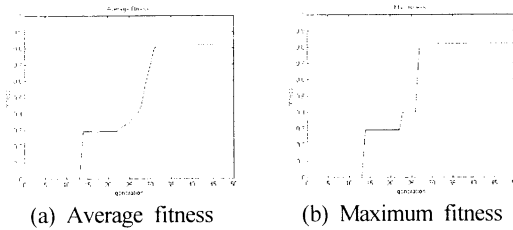

(a) Average fitness (b) Maximum fitness
Fig. 6: Trend of fitness values for robot control

Fig. 6 shows a trend of fitness values for robot control. Theoretical Maximum fitness is 1 by eq. (5) but s can not be zero in normal navigation path. Therefore, fitness cannot reach 1. After 50 generation, we obtain a $c = 0$, $p = 54$, $f(c, s) = 0.82$.
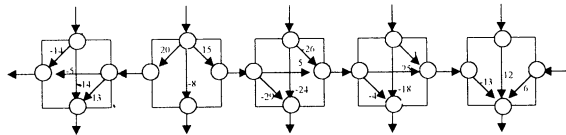

Fig. 7: Final structure and weights for BNN

Fig. 7 shows a final structure and weights for BNN. We obtain a simple BNN that is not used any bias term. The obtained BNN has a high fitness by genetic procedure.
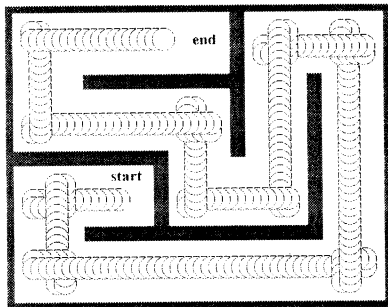

Fig. 8: Movement of robot with maximum fitness individual in training path

Fig. 8 shows a navigation and obstacle avoidance by best individual during whole genetic procedure. All movement about cornering of robot is not quite nice because of a wide width of path.
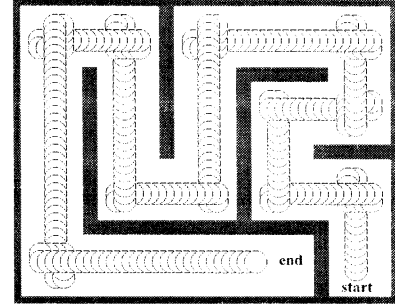

Fig. 9: Movement of robot with maximum fitness individual in test path

Fig. 9 shows a movement of robot with maximum fitness individual in test path. A robot shows exactly the same way of forward movement and cornering feature in Fig. 8. It means that BNN optimized in training path operate in other navigation path very well.

5. Conclusion

This paper propose a BNN that is designed to contain some characteristics such as easy hardware implementation, less complexity for structure optimization, and searching for global optimum. Structure and weights of BNN is optimized by genetic algorithm. For optimization of BNN, two-dimensional encoding is made use of a structure and weights simultaneously. In reproduction, modified scaling and elitist is used. Normalized two-dimensional crossover operator for two-dimensional encoding and different mutation probabilities in structure and weight bit string is applied for optimization of BNN. For performance increasement, Internal copy operator is also used. In a simulation, we shows that BNN can successfully apply a robot control field. Based on this research, more various experiments with real khepera is required.

References

[1] M. Vittorio, "Genetic Evolution of the Topology and Weight Distribution of Neural Networks," *IEEE Trans. on Neural Networks*, Vol. 5, No. 1, pp. 39-53, 1994.

[2] T. Kumagai, M. Wada, S. Mikami, and R. Hashimoto, "Structured Learning in Recurrent Neural Network Using Genetic Algorithm with Internal Copy Operator," *Proc. IEEE International Magnetics Conf.*, pp. 651-656, 1997.

[3] B. Thang Nguyen and M. Byung Ro, "On Multi-Dimensional Encoding/Crossover," *Proc. 6th International Conf. on Genetic Algorithms(ICGA)*, pp. 49-56, 1995.

[4] T. Bäck, U. Hammel, and H.-P. Schwefel, "Evolutionary Computation: Comments on the History and Current State," *IEEE Trans. on Evolutionary Computation*, Vol. 1, No. 1, pp. 3-17, 1997.

[5] M. Francesco and F. Dario, "Robotics and Autonomous Systems," Vol. 16, pp. 183-195, 1995.

From Life to Robotics: Social Robots

S. Reimann
Institute for Autonomous intelligent Systems
German National Research Center for
Information Technology **GMD**
Birlinghoven, Germany.
email: stefan.reimann@gmd.de

A. Mansour
BMC Research Center
The Institute of Physical and
Chemical Research (**RIKEN**)
Nagoya, 463-003, JAPAN
email: mansour@nagoya.riken.go.jp

Abstract

The formation of groups is of essential importance in biology. Aggregation is based on a simple dynamical model for the motion of each individual. It is shown that a *swarm* is formed within a short time span and is stable under perturbations.

Keywords: *Grouping behavior, stochastic control.*

1 Introduction

Following the work by E. Haeckel, multicellular organisms had evolved from *volvox-like colonies* of unicellular, autonomously acting organisms. The aggregation of *Dictyostelium* cells to highly organized colonies is one of the best studied examples for this kind of self-organization in ensembles of interacting individuals. Aggregation may be induced by environmental changes. Under gentle environmental conditions, *Dictyostelium* cells live alone, while if, for example, nutrance becomes rare, 3-dimensional colonies are constituted, exhibiting a high degree of functional organization up to differentiation [1]. Higher organisms show grouping behavior as well: Particular tasks like hunting or defense against enemies may not be achievable by single individuals, but may be realizable by cooperation of many. In general, one may claim

that the maintenance of the group, not necessarily the maintenance of single individuals, may be regarded as the most fundamental function of social behavior. As such the establishment of groups is of vital importance.

The formation of colonies has to be sufficiently fast and must be stable with respect to external disturbances. These particular features will be more closely studied in the following. The aim of this work is not to mimicke any particular biological phenomena, rather than to discuss these features from a principal point of view. To make this point clear, we call the objects under consideration agents. Our model of an *agent* is based on biological considerations [2] about simple organisms such as protozoa, bacteria, up to insects. We believe that the same functional architecture is analogously realized in higher organisms as well.

2 Restricted random walk

Motion patterns of simple organisms are commonly regarded as generated by random mechanisms. Accordingly, we base our model on a "random walk"-like mapping

$$x \mapsto \varphi(x), \qquad x \in X, \qquad (1)$$

where $x \in X$ is the agent's position in some space X, and φ is a random mapping on X. We assume

that the agent at each time approximately maintains its former direction, i.e.

$$0 \leq \frac{\langle x, \varphi(x) \rangle}{\|x\| \, \|\varphi(x)\|} \leq 1 - \alpha,$$

where $0 \leq \alpha \leq 1$ may be interpreted as the "flexibility" of the agent. Obviously, if $\alpha = 0$, then the motion of the agent is deterministic. Equivalently, we assume that the determinant of the linearized mapping associated with φ is non-negative. If it is zero, the agent is stationary. The motion due to (1) is a (restricted) random walk having trajectories of bounded mean curvature, determined by α.

If the motion of an agent would be due to a random walk, the mean time needed to reach a given target would be a quadratic function of the agent's initial distance from that point. Hence, it would take a quite long time to reach a point from a far distance. Moreover, as also known from the Theory of Diffusion Processes [3], the agent will come close to any given point only if X does not have a dimension larger than 2. In three dimensions it will not. Consequently purely diffusive motion does not provide a reliable strategy to sufficiently quickly form groups.

3   Weighted random Walk

The above elementary model can be extended as follows. As already stated in the basic model the determinant of the motion generating mapping is non-negative. The only extension consists in multiplying φ with a weight taking values in the set of real numbers. This weight depends on the actual motion of the agent, i.e. as a function, the weight depends on the agent's actual position and its position before. For details about the model and in particular about the definition of the weight function, see [4]. The motion is generated by the following mapping defined on $X \times X$ having its range in X

$$(x, x') \longmapsto \varphi_{c(x,x')}(x), \qquad (2)$$

where x' is the precursor of x. As such the determinant of the linearized mapping $\mathcal{L}_c$ corresponding to φ_c can be either $+1$ or -1 depending on the value of $c(x, x')$. Accordingly, the agent either continues or reverses its former direction.

This simple extension leads to fundamentally new motion features: Firstly, it can be shown [5] that the agent will reach a given point almost surely in all dimensions, in particular in those larger than 2. Secondly, its mean arrival time is only a linear function of its initial distance. Consequently, reaching a giving point happens much faster than in a purely diffusive setting. In summary, motion due to a weighted random walk as defined above provides reliable and fast target-reaching behavior.

4   Following behavior

The first step in analyzing group behavior is to consider only two agents. As our scenario we consider a "prey" moving along some given path, while it is hunted by a "predator". We assume that the motion of the prey is deterministic, while that of the predator is due to a weighted random walk. The question is: Will the predator reach its prey? The answer, roughly speaking is "Yes", if the predators speed is not too low compared with that of its prey. A second setting is that the prey moves completely randomly with some mean escape time. Again, the predator will reach the prey if its relative velocity is high enough. This stable reaching behavior is due to the stability of the dynamics generated by a weighted random walk, as defined above.
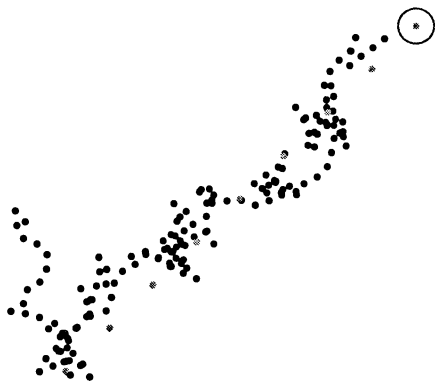
Figure 1: One robot moving along a nearly straight line (deterministic), while a second one follows it by our algorithm.

5 Grouping behavior

The generalization to an ensemble of n interacting agents is straight forward. For simplicity we may assume that interaction is distance-dependent, such as in interactions mediated by light, sound, or based on diffusing chemicals. As such, the motion of a single agent depends on the motions of all other agents in the group. Again we avoid to define the corresponding mapping generating the motion of an agent in a group of signalling agents in detail; however the motion is a function $\Phi_{\mathbf{c}} : X^{2n} \to X^n$ defined by

$$(\mathbf{x}, \mathbf{x}') \longmapsto \Phi_{\mathbf{c}(\mathbf{x}, \mathbf{x}')}(\mathbf{x}), \qquad (3)$$

where $c_i(x_i, \mathbf{x}')$ is the weight of the i-th agent depending on the motion of all other agents in the group. As above, $\mathbf{x}'$ is the precursor of $\mathbf{x}$. It can be shown that under fairly mild assumptions this mapping has a stable fixed point in X^n. This point corresponds to a stable configuration of the agents. If all agents are identical, the asymptotic configuration is a *finite* "cloud".

The time needed for establishing such a cloud also is a linear function of the initial distances (see [5]). Hence the formation of such groups can happen quite quickly. Because of stability, this cloud will be maintained under perturbances. In particular, if one agent moves along a given line without respect to influences by other agents, the ensemble will follow it. This can be regarded as elementary swarming behavior.
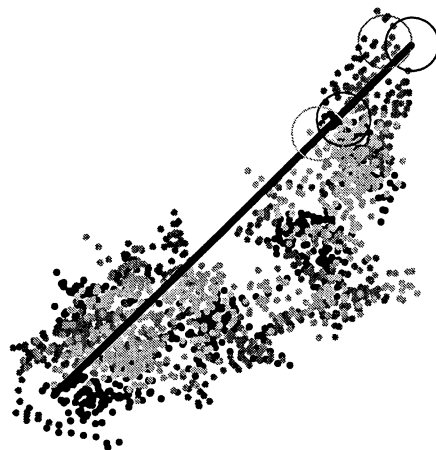
Figure 2: One robot moving along a nearly straight line (deterministic), while the rest (the cloud) follows it.

6 Summary

In this paper we have considered an elementary extension of a model generating random-like motion in space. It was shown that target-reaching was reliably achieved within a time-span, being less than that expected in purely random systems. As a consequence, stable following behavior was obtained, as well as the formation of groups of interacting agents. The computational effort needed is minimal and, hence, the performance is fast. Models based on "weighted random walks" may provide both, a general

framework for modeling basic motion features of simple organisms, as well as a methodology to reliably control simple robot systems.

References

[1] J. D. Murray, *Mathematical Biology*, Springer Verlag, 1989.

[2] S. Reimann and A. Mansour, "Oriention by weighted randomness," *Artificial Life and Robotics*, vol. 4, 2000, To appear.

[3] J.L. Doob, *Stochastic Processes*, Wiley Publications in Statistics, 1953.

[4] S. Reimann and A. Mansour, "Navigation by weighted chance," in *The Second International Conference on Intelligent Processing and Manufacturing of Materials IPMM'99*, July 10 - 15 1999, pp. 1103 – 1107.

[5] S. Reimann and T. Krüger, "Reliable motion control," In preparation.

Viewing Autonomous Robot Design as Self-Organizing Complex Dynamical Systems

A. D'Angelo
Dip. di Matematica e Informatica
Udine University
Udine (I), I-33100

F. Montesello
Dip. di Elettronica e Ingegneria
Padua University
Padova (I), I-35100

E. Pagello
Dip. di Elettronica e Ingegneria
Padua University
Padova (I), I-35100

Abstract

The behaviour-based approach deals with robot operations, inside an unstructured and partially unknown environment, as they were accomplished thinking at the robot as a collection of interacting parts, its behaviours, competing for common resources, its sensors and actuators. The challenge to provide a qualitative depiction of autonomous robot behaviour, performing either individual or collective tasks, has suggested us to consider robots as open systems that receive a stream of input sensor signals from and generate a stream of output actions to the environment. Using the framework of complex dynamical systems we have introduced the concepts of *energy* and *entropy* with the aim to make emerging a *collective behaviour* from a robot team as if it were caused by specific patterns of self-organizing groups.

1 Introduction

During the last decade the behaviour-based approach has come to maturation as a design principle for autonomous robot control which, starting from the pioneristic work of Brooks [4], gave rise to several innovative architectures such as Connell[5], Maes [9], Arkin [2], Donath [1], Pfeifer [14] D'Angelo [6], Kaelbling [8].

The driving force of this approach can be found in what all authors have pointed out how their systems are built upon a number of component parts whose reciprocal interactions play a crucial rôle. With the same respect, the embodiment of situated agents have gained a central concern because it alleviates a number of problems which arise when a disembodied intelligent system is considered.

Furthermore, in our experience of designing the distributed control ([11], [10], [12], [13]) of an agent team playing a simulated soccergame for the RoboCup competition, we have been faced with problems involving specified relations over the interaction dynamics of groups of agents, their intention to cooperate and under which conditions and so on.

Such considerations, along with the recent results on autonomous agents, emphasize how the language of complex dynamical system theory can provide a framework where *non-linearity, interaction* among component parts, keeping *far from equilibrium* while *dissipating energy* play a central rôle.

In the rest of the paper we shall outline some basic properties of dynamical system theory where autonomous agents are considered open systems that receive a stream of input sensor signals from and generate a stream of output actions to the environment. To simplify the discussion we have restricted our investigation to robots moving on a plane, called **roboticles**, defining them as abstract autonomous vehicles with two motor wheels, driven by the speed and the compass of their *moving center* while receiving input data from their *virtual sensors*.

2 Dynamical Systems

If we want to emphasize the interaction with the world, the best perspective is that of looking at a robot as an open dynamical system, where a set of *state variables* $\mathbf{x}$ and a *dynamical law* Φ are given, telling how the values of the state variables change over time. In the continuous-time case the dynamical law takes the form of a set of differential equations referred to as a *vector field* on the state space. For a more complete survey on this topic you can see Hale and Kocak [7].

Thus, a continuous non linear system with two state variables takes the form

$$\dot{x} = u(x, y, \lambda) \qquad \dot{y} = v(x, y, \lambda) \qquad (1)$$

where x and y are state variables whereas λ is the control parameter. We call the *trajectory* of the system the set of states generated by the action of the dynamical law moving from the *initial state* $< x_0, y_0 >$. Because the trajectory has the property that its tangent at each point is equal to the value of the vector field at that point, we can interpret u and v as components of the *velocity field* of a fluid, motivating the term *flow* to the set of trajectories through every point in the state space.

2.1 Long Term Behaviour

When we ascribe particular dynamics to control an autonomous robot we could be aware if its trajectory is diverging to infinity, namely, the robot is running away from a given position without restriction. The most interesting case, however, is the convergence to some *limit set*, defined as a set of points which are invariant with respect to the dynamical law.

A limit set is said *stable*, or an *attractor*, if it has the property that the trajectories passing through all nearby states converge to it. This means that a small enough perturbation, which moves the system away from the attractor, causes the dynamical law to bring the state of the system back to the attractor. The open set of initial states that converge to a given attractor is said *basin of attraction*.

A limit set is termed *unstable*, or a *repellor*, if it has the property that at least some nearby trajectories diverge from it. So, if the state of the system is perturbed even an infinitesimal distance from the repellor, the dynamical law will cause the system to carry it away.

A limit set which consists of only one *equilibrium point* is termed *node*, *focus* or *saddle point* depending on the form of trajectory shown by the system in the neighbour of the equilibrium point. Of course, this is not the only kind of limit set exihibited by a dynamical system: *periodic*, *quasiperiodic* and *chaotic attractors* are also possible.

While the first, also termed *limit cycle*, has a very simple structure of closed trajectories in the state space, and it corresponds to periodic or oscillatory solutions, the remaining two present a very complicated shape. Specifically, the last displays a fractal geometry which is a consequence of the sensitive dependence of the system from the initial conditions.

In general, the state space of a dynamical system will contain multiple attractors, each surrounded by its own basin of attraction, separated by unstable manifolds. The global characterization of this cellular structure is called *phase portrait*.

2.2 Energy

With respect to the dynamical law *(1)*, many different classes of dynamical systems can be identified, depending on the specific properties one would emphasize. In the case of autonomous vehicles, however, a more general definition of energy provides a meaningful point of view. Consider, for example, a vehicle following the trajectory which results from a given dynamical law. The differential form *(2)* can be easily

obtained from *(1)*

$$v(x, y, \lambda)dx - u(x, y, \lambda)dy = 0 \qquad (2)$$

stating that the vehicle trajectory is always parallel to the *vector field*. Generally, *(2)* is not an exact differential form, even if an integrating factor can be found by which it becomes exact. So, choosing $\frac{1}{E}$ as such a factor, we can rewrite *(2)*

$$\frac{v}{E}dx - \frac{u}{E}dy = d\Psi \qquad (3)$$

where $\Psi(x, y)$ denotes a scalar function which implicitally defines a possible trajectory traced by the vehicle in according to the given initial conditions.

Now, it can be easily shown that *(3)* holds if and only if the following differential equation holds

$$u\frac{\partial E}{\partial x} + v\frac{\partial E}{\partial y} = 2\sigma E \qquad (4)$$

which allows us to interpret *E(x,y)* as the **energy** a vehicle has while it is moving along its trajectory. You should notice that σ represents the *dissipative factor*, defined as an half of the divergence of the vector field.

Equation *(3)* provides a general way to build the dynamical law of a vehicle

$$\dot{x} = -E\frac{\partial \Psi}{\partial y} \qquad \dot{y} = E\frac{\partial \Psi}{\partial x} \qquad (5)$$

as soon as a trajectory family $\Psi(x, y)$ and an energy function $E(x, y)$ have been assigned. If $E(x,y)$ is a constant then *(2)* is an exact differential form, so, we say the system to be *hamiltonian* or *conservative*.

2.3 Effort

Let us now define the product Vds as the **effort** delivered by the vehicle effectors to support its movement, along a trajectory arc ds with speed V. Using the components u and v of the vector field we can rewrite the preceding definition with the inner product

$$Vds = udx + vdy \qquad (6)$$

so it can be easily shown that, if the cross derivatives are equal, the amount of effort spent by the effectors to move a vehicle along its trajectory only depends on initial and final positions. In such a case there exists a scalar function $F(x, y, \lambda)$ which generates the vector field components as partial derivatives

$$\dot{x} = -\frac{\partial F}{\partial x} \qquad \dot{y} = -\frac{\partial F}{\partial y} \qquad (7)$$

The main feature of these systems lies in the way their attractors and repellors are located, simply looking at the distribution of minima and maxima of the scalar function F, which provides easily identifiable basins of attraction.

The neat effect of the dynamical law is always a decrement of the value of F while the system is travelling to some attractor, so repellors, which correspond to *local maxima*, cannot ever be reached. Because its value can be used to estimate how much residual *energy* must be dissipated before reaching an attractor, it motivates the term *dissipative* for the function F. Moreover, the negative laplacian $-\nabla^2 F$ equates the double of the dissipative factor.

3 Roboticles

The dissipative dynamics *(7)* is a simple but powerful method to solve the navigation problem. When we want an autonomous vehicle to reach a given target, we implement a governor unit which builds incrementally, using sensor information, a *dissipative function* of additive terms, shaping the landscape of recognized objects, with the property of generating a repulsor, if obstacle avoidance is requested, and an attractor for the identified target.

The principal drawback of this approach is that the vehicle will stop as soon as the given task has been accomplished because the initial amount of energy, delivered to the vehicle in the initial position will be dissipated while it is moving to the target. So, if we want to keep such a system far away from equilibrium we need to supply it with a new amount of energy. To this aim it is sufficient to provide a vehicle with a mechanism able to acquire energy from outside.

3.1 Moving Vehicles

Within this framework we have devised a dynamical system model for autonomous vehicles, that we have termed **roboticles**, where sensor data are used to increment their energy level which is dissipated later by the effort released by their effectors.

This model, however, requires the system to be open with respect to energy and this objective is achieved by making explicit in *(1)* the contribution of the environment to the vehicle dynamics. The following equations

$$
\dot{V} = [\sigma + \gamma_1 \cos(2\vartheta) + \gamma_2 \sin(2\vartheta)]V
$$
$$
\dot{\vartheta} = \omega + \gamma_2 \cos(2\vartheta) - \gamma_1 \sin(2\vartheta)
$$
(8)

solve the problem because they introduce the state variables V and ϑ with respect to the *moving center* of the vehicle.

The new dynamical law *(8)* has been obtained from *(1)* as time derivatives, considering that the speed V and the stearing ϑ define a vector tangent to the vehicle trajectory. So, the coefficients appearing in *(8)* take the values

$$
\begin{array}{ll}
\sigma = \frac{1}{2}\left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right] & \omega = \frac{1}{2}\left[\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right] \\
\gamma_1 = \frac{1}{2}\left[\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right] & \gamma_2 = \frac{1}{2}\left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right]
\end{array}
$$
(9)

which show how the contribution of the vector field components are brought about in driving the vehicle motor wheels whose control can be easily derived from the state variables.

The preceding equations, however, are a particular case of the most general one, appearing below

$$
\dot{V} = \alpha(V, \vartheta, p_1, \ldots, p_m, \lambda)
$$
$$
\dot{\vartheta} = \zeta(V, \vartheta, p_1, \ldots, p_m, \lambda)
$$
$$
\dot{x} = V \cos\vartheta - \frac{\partial G}{\partial x}
$$
$$
\dot{y} = V \sin\vartheta - \frac{\partial G}{\partial y}
$$
(10)

with the state variables V and ϑ for the vehicle and the position variables x and y ascribed to the environment, expressing the necessary embodiment, and used to locate the vehicle during the simulation phase. The last two equations assume the vehicle moving on a generic surface, whose features are defined by the function G, which becomes a perfectly rigid plane when G is taken as a constant.

3.2 Breitenberg Vehicles

Several properties can be obtained from the definition of roboticles and which can be used to simulate many situations arising with both single and multi-robot framework. The simplest situation is that described by Breitenberg [3] within his vehicle *I*. Using the results presented in the preceding sections it can be easily shown that, for any given energy function E, equations *(1)* take the form

$$
\dot{r} = -\frac{E(r)}{r} \qquad \dot{\varphi} = 0
$$
(11)

using a polar frame of reference, centered on the source of light which drives, modifying its energy level, the behaviour of the vehicle.

The dynamicle law of the corresponding roboticle is immediately obtained as time derivative with $\dot{r} = V$

$$\dot{V} = -p(r)V \qquad \dot{\vartheta} = 0 \qquad (12)$$

where $p(r)$ is the vehicle virtual sensor and it is defined by the relation

$$p(r) = \frac{\partial}{\partial r}\left(\frac{E(r)}{r}\right) \qquad E(r) = r\frac{\partial F}{\partial r} \qquad (13)$$

whereas the dissipative function F takes the form resulting from the given differential equation.

4 Conclusions

Referring to the concept of *energy* we have proved that the dynamical law of a roboticle can be easily derived from both a given trajectory family and an energy function which is used as a surface where to put each trajectory. Moreover, the flow of data received by its sensors can be used to move a roboticle on this surface, eventually incrementing its energy level.

In the case of a group of interacting roboticles, performing the same task, the problem of generating a collective behaviour is reduced to the choise of a proper *interaction law*, which gives raise to *self-organization* as an aggregate of individuals working as a higher level unit. To force such a behaviour, however, we need to supply the group with an appropriate amount of energy. This means that each roboticle dynamical law shows a *bifurcation point*, appearing as a new pattern at the collective level.

Aknowledgement

References

[1] T.L. Anderson and M. Donath. *Animal behaviour as a paradigm for developing robot autonomy*. The MIT Press, Cambridge (MA), 1990.

[2] R. C. Arkin. *Integrating Behavoural Perceptual and World Knowledge in Reactive Navigation.* The MIT Press, Cambridge (MA), 1990.

[3] V. Breitenberg. *Vehicles, experiments in synthetic psychology.* MIT Press, New York, 1996.

[4] R. Brooks. A layered intelligent control system for a mobile robot. *IEEE J. Robotics Automat.*, RA-2:14–23, April 1986.

[5] J. H. Connell. *Minimalist Mobile Robotics.* Number 5 in Perspective in Artificial Intelligence. Academic Press, 1990.

[6] A. D'Angelo. *Using a Chemical Metaphor to Implement Autonomous Systems*, volume 992 of *Lecture Notes in Artificial Intelligence.* Springer, Florence (I), 1995.

[7] J. Hale and H. Kocak. *Dynamics and bifurcations.* Springer-Verlag, New York, 1991.

[8] L. P. Kaelbling and S. J. Rosenschein. *Action and planning in embedded agents.* The MIT Press, Cambridge (MA), 1990.

[9] Pattie Maes. *Situated agents can have goals.* The MIT Press, Cambridge (MA), 1990.

[10] E. Pagello, A. D'Angelo, F. Montesello, and C. Ferrari. Emergent cooperative behaviour for multirobot systems. In Y. Kakazu, M. Wada, and T. Sato, editors, *Intelligent Autonomous Systems (IAS98)*, pages 45–52, Supporo (Japan), June 1-4 1998. Ios Press.

[11] E. Pagello, A. D'Angelo, F. Montesello, and C. Ferrari. Implicit coordination in a multi-agent system using a behaviour-based approach. In *Distributed Autonomous Robot Systems 4 (DARS98)*, Karlsruhe (D), 1998.

[12] E. Pagello, A. D'Angelo, F. Montesello, F. Garelli, and C. Ferrari. Cooperative behaviors in multi-robot systems through implicit communication. *to appear in Robotics and Autonomous Systems*, 1999.

[13] E. Pagello, C. Ferrari, A. D'Angelo, and F. Montesello. Intelligent multirobot systems perfoming cooperative tasks. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, page to appear, Tokyo (Japan), 12-15 Oct. 1999.

[14] R. Pfeifer. Building fungus eaters: Design principles of autonomous agents. In P. Maes, M.J. Mataric, J.A. Meyer, J. Pollack, and S.W. Wilson, editors, *From Animals to Animats 4, Simulation of Adaptive Behaviour (SAB96)*, pages 3–12, 1996.

Experiments of a Floating Underwater Robot with 2 Link Manipulator

S. Sagara   T. Tanikawa   M. Tamura   R. Katoh
Department of Control Engineering
Kyushu Institute of Technology
Tobata, Kitakyushu 804, Japan

Abstract

In this paper, for a free floating underwater robot with 2 dimensional and horizontal planar 2 link manipulator we derive the dynamic model including the hydrodynamic forces and validate the effectiveness of the model from the simulation and experiment. Furthermore, we show an experimental result using a resolved acceleration control method. Experimental results show the effectiveness the model and the control method.

1   Introduction

Since underwater robots are necessary to ocean exploration, many studies have been done about Underwater Robotic Vehicle (URV) including Remotely Operated Vehicles (ROVs) and Autonomous Underwater Vehicles (AUVs) [1]. To get resources and construct underwater structures, URVs, especially AUVs, with manipulators are required, so high accurate dynamic model and/or control methods of manipulator are dominant to maintain a good control performance of AUV. Furthermore, the manipulators become large and the dynamics cannot ignore. Recent years some dynamic models of underwater manipulators have been proposed [2, 3, 4, 5, 6], and only a few experiments have been done to verify the accuracy of the model by using 1 link arm. The dynamic model of underwater manipulators contains nonlinear hydrodynamic forces such as added mass force and drag force, and the complete hydrodynamic model cannot be obtained. Therefore, robust control methods for the underwater manipulators are necessary to obtain the high control performance.

In this paper, for a free floating underwater robot with 2 dimensional and horizontal planar 2 link manipulator shown in Fig. 1 we derive the dynamic model including the hydrodynamic forces and validate the effectiveness of the model from the simulation and experiment. Furthermore, we show an experimental re-
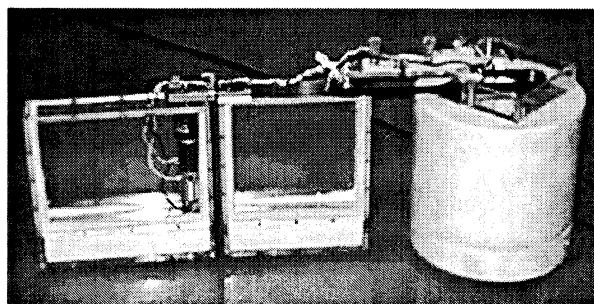

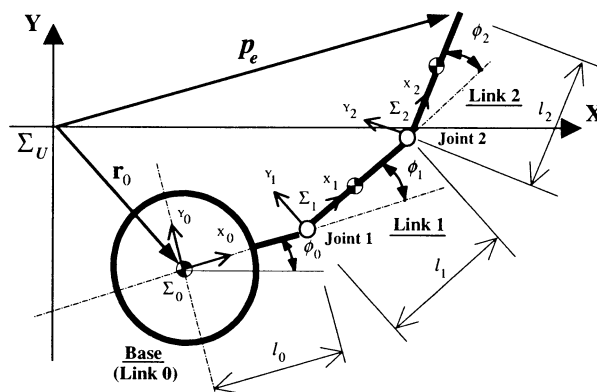Fig. 1: Floating 2-link underwater robot manipulator

Fig. 2: 2-link underwater robot manipulator model

sult using a resolved acceleration control method.

2   Kinematics and Dynamics

2.1   Modeling

The underwater robot model used in this paper is shown in Fig. 2. It has a robot base and 2-DOF manipulator which can move in a plane.

Symbols used in this paper are defined as follows:
[Symbols]
Σ_U : inertial coordinate frame

Σ_i : ith link coordinate frame ($i = 0, 1, 2$; link 0 means base)

$^U\!R_i$: coordinate transformation matrix from Σ_i to Σ_U

r_0 : position vector of origin of Σ_0 with respect to Σ_U

l_i : length of link i

v_i : velocity vector of link i with respect to Σ_i

ϕ_i : relative joint angle

ω_i : joint angular velocity ($= \dot{\phi}_i$)

θ_i : absolute joint angle

p_e : position vector of end-tip with respect to Σ_U

p_0 : position vector of center of gravity of base with respect to Σ_U

X_e : position and attitude vector of end-tip with respect to Σ_U ($= [p_e^T, \theta_2]^T$)

X_0 : position and attitude vector of center of gravity of base with respect to Σ_U ($= [p_0^T, \theta_0]^T$)

ϕ : joint angular velocity vector ($= [\phi_1, \phi_2]^T$)

m_i : mass of link i (link 0 means the robot base)

M_{a_i} : added mass tensor of link i

I_i : inertia tensor of link i

I_{a_i} : added inertia tensor of link i

$\hat{x}_i$: position vector from joint i to joint ($i + 1$) with respect to Σ_i

s_i : position vector from origin of Σ_i to center of gravity of link i with respect to Σ_i

E : unit matrix

2.2 Kinematics

First, a time derivative of the end-tip position and attitude vector X_e is

$$\dot{X}_e = A\dot{X}_0 + B\dot{\phi} \qquad (1)$$

where

$$A = \begin{bmatrix} 1 & 0 & a_{13} \\ 0 & 1 & a_{23} \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ 1 & 1 \end{bmatrix},$$

$$a_{13} = -l_0 S_0 + b_{11}, \quad a_{23} = l_0 C_0 + b_{21},$$
$$b_{11} = -l_1 S_1 + b_{12}, \quad b_{12} = -l_2 S_2,$$
$$b_{21} = l_1 C_1 + b_{22}, \quad b_{22} = l_2 C_2,$$
$$S_i = \sin\theta_i, \quad C_i = \cos\theta_i \quad (i = 0, 1, 2).$$

Next, let Q and L be a linear and an angular momentum of the robot including hydrodynamic added mass M_{a_i} and added inertia I_{a_i}, then

$$Q = [Q_1, Q_2, 0]^T = \sum_{i=0}^{2} {}^U\!R_i(m_i E + M_{a_i})\dot{s}_i, \quad (2)$$

$$L = [0, 0, L_3]^T = \sum_{i=0}^{2}(I_i + I_{a_i})\omega_i$$
$$+ \sum_{i=0}^{2} \hat{x}_i \times [{}^U\!R_i(m_i E + M_{a_i})\dot{s}_i] \qquad (3)$$

where

$$\dot{s}_i = v_i + \omega_i \times s_i, \quad \omega_i = [0, 0, \dot{\phi}_i]^T \ (i = 0, 1, 2).$$

From Eqs. (1), (2) and (3) the following equation can be obtained:

$$F = [Q_1, Q_2, L_3]^T = C\dot{X}_0 + D\dot{\phi} \qquad (4)$$

where $C \in R^{3\times3}$ and $D \in R^{3\times2}$ are matrices including the added mass M_{a_i} and the added inertia I_{a_i}. Here, we assume that the added mass and added inertia are constant. In reality, the added mass and inertia are variable but the influence of the variation is compensated by a control method in section 3.

2.3 Fluid drag force and moment

Generally, the drag force and moment of the joint i can be represented as follows [4, 5]:

$$f_{d_i} = \frac{\rho}{2}C_{D_i}D_i \int_0^{l_i} \|w_i\|w_i d\hat{x}_i, \qquad (5)$$

$$t_{d_i} = \frac{\rho}{2}C_{D_i}D_i \int_0^{l_i} \hat{x}_i\|w_i\|w_i d\hat{x}_i \qquad (6)$$

where $w_i = v_i + \omega_i \times \hat{x}_i$, and ρ is the fluid density, C_{D_i} is the drag coefficient, D_i is the height of link i.

2.4 Equation of motion

Considering the hydrodynamic forces described above and using Newton-Euler formulation, the following equation of motion can be obtained:

$$(M + M_a)\ddot{q} + b(q, \dot{q}) + F_D = [0, 0, 0, \tau_1, \tau_2]^T \quad (7)$$

where $q = [X^T, \phi^T]^T$, and M is the inertia matrix, $b(q, \dot{q})$ is the vector of Coriolis and centrifugal forces, M_a is the matrix consisting of the added mass and inertia, F_D is the vector consisting of the drag force and moment, τ_i is the joint torque.

3 Resolved Acceleration Control

To design a position control system of the end-tip, a resolved acceleration control method is used in this paper.

Differentiating Eqs. (1) and (4) with respect to time, the following equation can be obtained:

$$\begin{bmatrix} \ddot{p}_0 \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} C & D \\ \tilde{A} & \tilde{B} \end{bmatrix}^{-1} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \qquad (8)$$

where

$$z_1 = \dot{F} - \left(\dot{C} \dot{X}_0 + \dot{D} \dot{\phi} \right), \quad z_2 = \ddot{p}_e - \left(\ddot{\tilde{A}} \dot{X}_0 + \dot{\tilde{B}} \dot{\phi} \right),$$

$$\tilde{A} = \begin{bmatrix} 1 & 0 & a_{13} \\ 0 & 1 & a_{23} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}.$$

If the values of the hydrodynamic forces are exact, using Eq. (8) and the desired value of the end-tip p_d, the desired acceleration value of the joint $\ddot{\phi}_d$ can be obtained. However, the complete hydrodynamic model cannot be obtained. Furthermore, we assume that the added mass and inertia are constant, and the drag force and moment are approximated in this paper.

Therefore, instead of $\ddot{p}_d$ the following modified desired value of the end-tip $\ddot{p}_d^*$ is utilized:

$$\ddot{p}_d^* = \ddot{p}_d + K_v(\dot{p}_d - \dot{p}_e) + K_p(p_d - p_e) \qquad (9)$$

where p_d is the desired position of the end-tip, and K_v and K_p are velocity and position feedback gains, respectively.

4 Experiment

4.1 Experimental system

Fig. 3 shows a configuration of experimental system. A robot that has a 2DOF manipulator is floating water. Each joint is actively rotated by a DC servo actuator consisting of DC servo motor, harmonic drive gear and incremental type encoder but no actuator for attitude control of the robot base. Physical parameters of the underwater floating robot is shown in Table 1.

Measurement and control system are consist of CCD camera, Video Tracker and 32bit personal computer. Two light emitting diodes (LED) is attached to the manipulator, and their motion is monitored by a CCD camera hanging from the ceiling. Video signals of the LED markers are transformed into the position data by Video Tracker, and put into a personal computer via a GPIB communication line. Using the position data and rotational angle of each joint measured by the incremental type encoder, the positions and attitude angles of the robot base and manipulator are computed in the personal computer that is also used in controller.
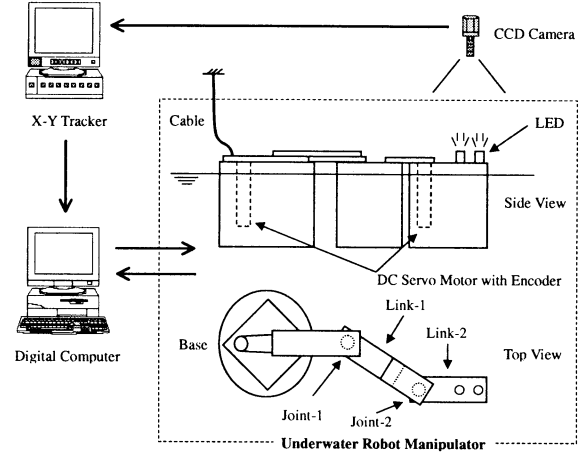

Fig. 3: Configuration of underwater robot system

Table 1: Physical parameters of the underwater robot

|	Base	Link 1	Link 2
Mass [kg]	31.72	7.48	9.68
Moment of inertia [kg m²]	0.41	0.20	0.37
Link length [m]	0.16	0.30	0.30
Height [m]	0.30	0.30	0.30
Added mass [kg]	48.26	32.82	32.82
Added moment of inertia [kg m²]	0	0.15	0.33

4.2 Verification of model

To verify the accuracy of the model, open loop simulation and experiment are performed. In the case of simulation, the drag coefficients are constant values $C_{D_0} = 1.12$, $C_{D_1} = C_{D_2} = 2$.

Fig. 4 shows an example of the results that joint 1 moves at a constant rotational velocity. In this figure, From the figure it can be seen that the simulation and experiment are slightly different because of the influence of the approximated hydrodynamic forces. Furthermore, in reality, C_{D_i} is not constant. However, the behavior is similar and it is considered that the model is appropriate.

4.3 Experimental result

The typical experimental result of the end-tip position control is shown in Fig. 5 and Fig. 6. In this experiment the reference trajectory of the end-tip is straight line. The feedback gains are $K_v = 2$ and $K_p = 1$. Fig. 5 shows the motion of the robot. In the other hand, Fig. 6 shows the time history of the de-
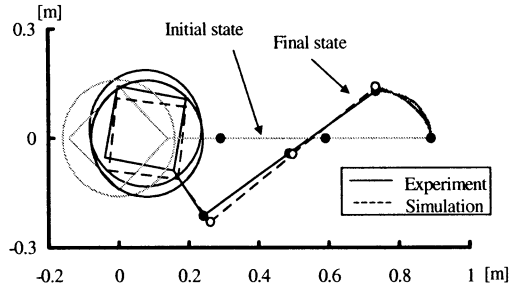
(a) $\phi_1 = 15$ deg/s
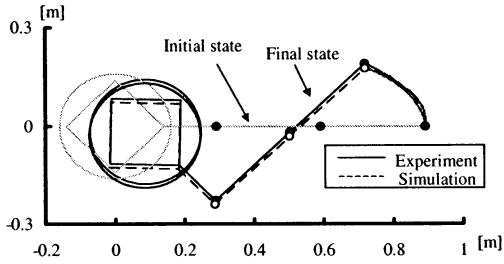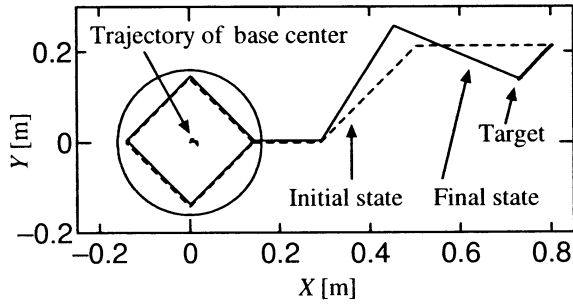

(b) $\phi_1 = 30$ deg/s

Fig. 4: Open loop result

Fig. 5: Motion of the underwater robot

Fig. 6: Tracking error

sired acceleration pattern and of the end-tip tracking error. In Fig. 6 the tracking error doesn't converge to 0 because of the influence of the waves. However the error is very small. From these figures it can be seen that the end-tip follows the reference trajectory in spite of the influence of the hydrodynamic forces and the resolved acceleration control method can be applied to the underwater robot manipulators.

5 Conclusion

We did experiments of a floating underwater robot manipulator. In spite of approximation of the hydrodynamic forces, experimental result showed the effectiveness of the resolved acceleration control method.
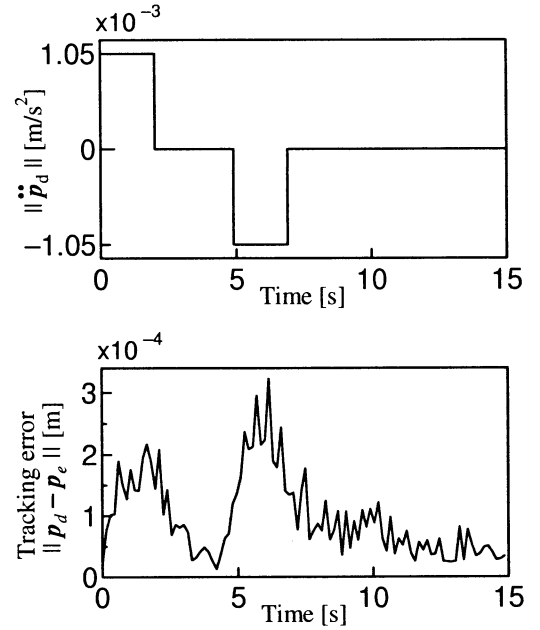
References

[1] J. Yuh ed., *Underwater Robotic Vehicles:Design and Control*, TSI Press, 1995.

[2] T. W. MaLain, S. M. Rock and M. J. Lee, "Experiments in the Coordinated Control of an Underwater Arm/Vehicle System", *Autonomous Robots 3*, Kluwer Academic Publishers, pp. 213-232, 1996.

[3] T. W. MaLain and S. M. Rock, "Development and Experimental Validation of an Underwater Manipulator Hydrodynamic Model", *Int. J. Robotics Research*, Vol. 17, No. 7, pp. 748-759, 1998.

[4] S. McMillan, D. E. David and R. B. McGhee, "Efficient Dynamic Simulation of an Underwater Vehicle with a Robotic Manipulator", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 25, No. 8, pp. 1194-1206, 1995.

[5] B. Lévesque and M. J. Richard, "Dynamic Analysis of a Manipulator in a Fluid Environment", *Int. J. Robotics Research*, Vol. 13, No. 3, pp. 221-231, 1994.

[6] T. J. Tarn, G. A. Shoults and S. P. Yang, "A Dynamic Model of an Underwater Vehicle with a Robotic Manipulator", *Autonomous Robots 3*, Kluwer Academic Publishers, pp. 269-283, 1996.

A Target Following Robot System Built on an Autonomous Mobile Platform

Koichiro ISHIKAWA and Akito SAKURAI

School of Knowledge Science

Japan Advanced Institute of Science and Technology, Hokuriku

1-1 Asahi-dai, Tatsunokuchi, Nomi, Ishikawa 923-1211, Japan

koichiro@jaist.ac.jp   asakurai@jaist.ac.jp

Abstract

We constructed on an autonomous mobile platform a robot system that avoids obstacles and follows its target, a walking person. We evaluated its efficiency by experiments in real environments commonly seen in daily life. This is a fist step toward an amenity robot which lives around us, will assist as well as entertain us, and enhance quality of our life.

Key Words: autonomous mobile robot, amenity robot, interactions with human, real environment.

1 Introduction

A new application field of robots is emerging in the last year, a pet or an entertainment robot [1][2]. Sony's AIBO is the one, and Furby may be another if one slacken the definition of robot. Limited ability or intelligence of the robot is found to be no obstacle to be a pet. We would expect from the fact that AIBO is widely accepted that in a near future amenity robots will assist as well as entertain us and enhance quality of our life.

As a very first step toward amenity robots we planned to build a robot that moves closely with a person, or more specifically follows a person. There are many functions and capability that amenity robots should have. They are divided into two categories: mental and physical. Following a person belongs to the latter and is one of the baby's intended and adorable actions of early developing stage and also one of the pet's basic required action. We did not intend to implement a mental function but we have realized that the person-following function has turned out to bring about a kind of attachment to the robot as we usually feel for a baby or a pet that follows us.

The amenity robot has to work in real environments and so does our experimental robot. It has to move around autonomously and so does ours. A pet robot need not to accomplish a complicated task (if it could do it would be better, though) but it must to be seen intelligent and adorable. As a natural consequence we have adopted behavior-based robot architecture.

The behavior-based robot paradigm is the one to decompose a problem into task achieving behaviors rather than into a series of functional units [3]. The amenity robots should work in real environments which is dynamic and unpredictable, operate on time scales similar to those of animals and humans, and be robust in the face of uncertain sensors [4].

2 Platform

We housed our system in an off-the-shelf platform, a multi-purpose autonomous mobile robot, ATRV-Jr that was designed and built by Real World Interface Division of IS Robotics, Inc. It is 55cm in height, 77.5cm in length, 64cm in width, and 50kg in weight [Fig.1].

We chose this platform to reduce developing cost and time. We also intended to figure out the ability and limitations of this platform. So far we have not added any hardware to the system.

ATRV-Jr has seventeen ultrasonic sensors (sonar) on its body, five sensors on the front, ten on the sides and two on the rear. By using these sensors our robot can detect and locate obstacles and a target around it.

ATRV-Jr runs with its four wheels at the speed of about 10km/h maximum. It makes a turn by a differential steering mechanism so it can change its direction while standing at a point. Two electric motors powered by two car batteries are powerful enough to run it in outdoor fields, rough and bumpy. ATRV-Jr is controlled by two commands: translate and rotate. The translate command specifies the direction, backward or forward, and the velocity of straight movement. The rotate command specifies the direction,
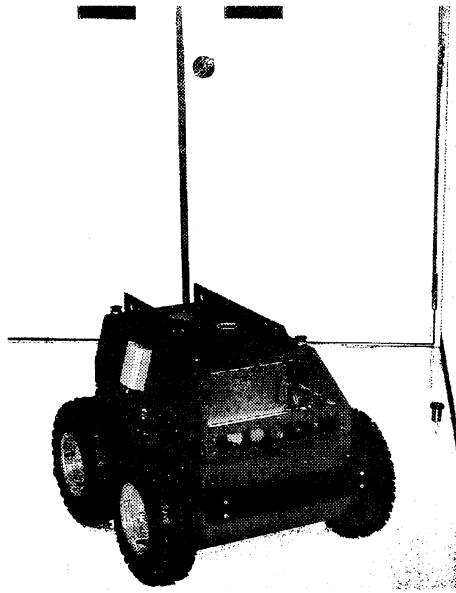
Fig. 1: ATRV-Jr is a mobile robot platform designed and built by Real World Interface Division of IS Robotics, Inc.

when it finds obstacles in a certain distance. Something that is not a target and in less than 125cm from the head of the body is considered to be an obstacle [Fig.2].
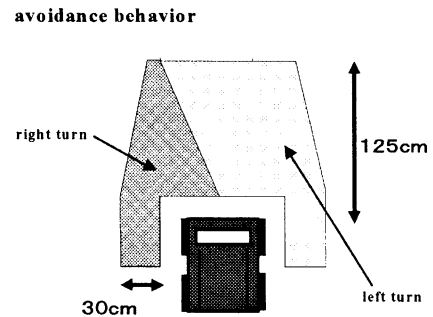
avoidance behavior

Fig. 2: One of the two types of avoidance behavior is selected according to the relative location of detected obstacle.

right or left, and the velocity of rotation.

ATRV-Jr has an on-board computer with a 350MHz Pentium II CPU. It can move around autonomously controlled by the computer as well as manually by a wired joystick or a remote computer connected by wireless or wired LAN.

3 Software

The software system we constructed consists of three independent basic modules, each of which executes a well-specified single behavior, prevention against collision, avoidance of obstacles and following a target. The module to be executed is selected by its priority calculated with respect to the situation where the robot is in. Our basic design paradigm is that of behavior-based robot.

To avoid collisions, we had better consider an envelope of the robot body to do without detailed calculation of contacts but still to be able to detect possible collisions with overhangs, e.g. tires or bumpers for ATRV-Jr. Our robot calculates approximate distance from its envelope to something by using sensor data. It stops its motion when it finds something close to it, 10cm from the envelope, for prevention measures.

Our robot enters into the "avoidance behavior"

When it finds an obstacle it turns left since it keeps the target on its left-hand side and it does not want to lose the target by turning right. If the obstacle is very close to its left-hand side it turns right, though.

Our robot follows a person at its left-hand side. It approaches the target when the target is apart about 100cm from its envelope, and departs from the target when the target is apart only about 30cm. This behavior is implemented by a very simple program. Our robot has an artificial vector field of motion velocity and directions. The filed is divided into thirteen subfields in a matrix form [Fig.3].

When the robot detects a target in the subfield A, its direction is set to about 30 degree left from ahead. Likewise if a target is in subfield B, the direction is set to about 60 degree right from ahead.

To accomplish the task mentioned above, coordination of the basic modules is indispensable. The most critical point is how to decide which module should be in charge since the two modules "avoidance" and "following" conflict with each other in complicated situation. The "following" module is activated when nothing but the target can be detected prespecified area and the "avoidance" module is activated otherwise.
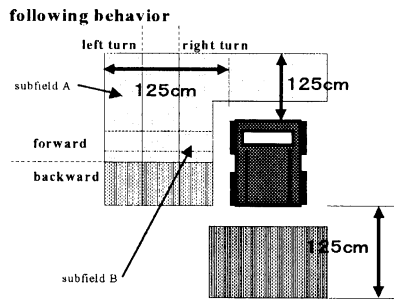
following behavior

Fig. 3: One of the thirteen sets of parameters for direction and velocity is selected according to the relative location of detected target.

4 Experiments

We selected for a location of the first experiment a slope designed for wheelchairs beside short steps at the front entrance of our school building. The slope is 120cm wide, goes downward at about 4 degrees, makes two left turn at the right angle, goes downward at a 4 degrees again and makes right turn at right angle to the exit of slope [Fig.4].
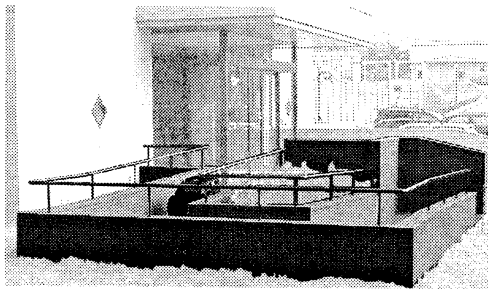

Fig. 4: The experimented slope and ATRV-Jr. It is too narrow for the robot to follow a target at its side.

Although the slope is narrow and has three right angle bents, it is only a way to follow since its alternative is the steps that the robot or a wheelchair cannot go down. The robot should follow a person at the person's right-hand side until they reach the slope, where the slope is too narrow for the robot to follow at its side. The robot should wait for a while to see the person goes down the slope a bit and then follow at the person's back. When the slope ends the robot should move faster to reach the person's right-hand side.

5 Results

We tested our robot at the slope mentioned above after a mock-up corridor in a laboratory, by which we tuned manually several parameters for the robot: distance to start an avoidance behavior and the vector field for target following.

At the real slope, we succeeded thirteen times in twenty trials. In the failed cases, four times are failures for entering the narrow slope, three times for turning right at the exit. There was no failure at making left turns on the slope.

We would like to add one observation which is a kind of unexpected result. When the robot follows us, the behavior is not straightforward. It seems sometimes to hesitate or to rush. Sometimes its movement is unexpected or as we expected. These behaviors made us feel that the robot is adorable, which would be one of the basics for amenity robots.

Note that the attractiveness is not caused by emergent emotion in the robot [6]. It exists not in the robot but in the human that operates the robot.

6 Discussion

We think that the results above are assuring for the correctness of our approach. The task itself is complicated and full of uncertainties. It is hard to build a robot to solve it based on a sequential decomposition paradigm [3]. We have overcome the difficulty by the behavior-based robot approach with small software and good success ratios. We have obtained another result that the robot behaves adorably, which is one of the keys to amenity robots.

There remain many problems, though. One is that we have several parameters which are difficult to tune. Another is a module-coordination program, which is by the way so small that we do not call it a module. The latter one would become much problematic when we add modules to the robot, which will increase complexity of coordination. These two could be solved by some learning methods for the former and by evolutionary methods for the latter [5].

Robot system with only the ultrasonic sensors has clearly limitations. First, the robot cannot distinguish a person from a wall. The robot is programmed so as to react to anything that is found to be at a specific position around the robot. At the experimented slope, the robot sometimes made several left turns to avoid a person by misunderstanding it to be a wall, and then to avoid the wall. Second, the robot has broad blind

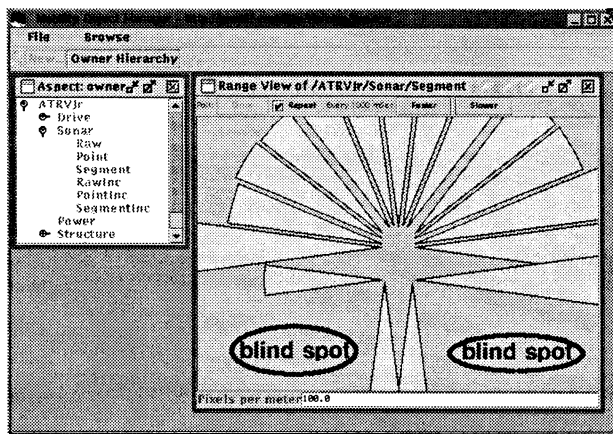spots in its rear side because sensors are mounted not in isotropic way [Fig.5].

Fig. 5: Range view of sonar sensors on ATRV-Jr by MOBILITY^{TM} distributed by Real World Interface Division of IS Robotics, Inc.

When a target person turns back in the narrow corridor, the robot cannot do anything but stand still. We cannot program to make a turn, because the corridor may be too narrow for the robot to turn back. Moreover programming the robot to go backward is to have a high risk of the robot's colliding with a wall.

7 Conclusion

We implemented a behavior-based robot system that follows a target person. We tested it in a real environment to follow a person walking from a wide area to a narrow corridor. The robot exhibits adorable behavior which would lead us to build amenity robots.

We have found some limitations of current implementation. We are planning to improve and enhance it in the following points:

1. robustness to variations of corridors and entrance to them,

2. robustness to variations of the direction or degree of slopes and the friction of ground, which is common to outdoor fields,

3. addition of sensors and sensor fusion mechanism: inclinometers and vision processing capability,

4. learning and evolution,

5. collaboration of robots, and

6. mental communication between the robot and humans.

Acknowledgement

References

[1] Kaneko K (1999), "Pet Robots (in Japanese)," *Nikkei Business*, July 12, 1999, pp.73-76.

[2] Harada M and Imai T (1999), "Start with toys first (in Japanese)," *Nikkei Electronics*, No. 747, July 12, 1999, pp.123-140.

[3] Brooks RA (1986), "A Robust Layered Control System for a Mobile Robot," *IEEE Journal of Robotics and Automation*, RA-2 Apr., 1986, pp.14-23.

[4] Brooks RA (1991), "Intelligence Without Reason," *Proc. of the 12th Int. Conf. on Artificial Intelligence (IJCAI-91)*, Sydney, Australia, Aug. 24-30, 1991, pp.569-595.

[5] Harvey I (1994), "Evolutionary Robotics and SAGA: The Case for Hill Crawling and Tournament Selection," *Artificial Life III*, Ed. Langton CG, SFI Studies in the Sciences of Complexity, Proc. Vol. XVII, Addison-Wesley, 1994, pp.299-326.

[6] Gomi T, Vardalas J, Ide K (1995), "Elements of Artificial Emotion," *Proc. of the 4th IEEE Int. Workshop on Robot and Human Communication (RO-MAN '95)*, Tokyo, Japan, July 5-7, 1995.

Eye-Gaze Tracking from Sequential Images

Do Hyoung Kim   Jae Hean Kim   Myung Jin Chung

Dept. of Electrical Engineering

Korea Advanced Institute of Science and Technology (KAIST)

bborog@cheonji.kaist.ac.kr, kjh@cheonji.kaist.ac.kr, mjchung@ee.kaist.ac.kr

Abstract –The main objective of this paper is to propose an efficient approach for real-time eye gaze tracking from a sequence of eye images. In the proposed system, eye features which are extracted by preprocessing the images are chosen as tracking parameters. To this end, we formulate an efficient model of the eye, using the gray level information of the eye image sequence. This image-based method is different from traditionally eye movement measuring systems, which are used by model-based algorithms, and is implemented using an inexpensive PC, fitted with a commercial-grade video camera and image frame grabber. The inputs to the system are images taken by a camera and these images are digitized by image grabber for subsequent computer processing. The measuring data are sufficient to describe the eye movement, because the camera is stationary with respect to the head. Experimental result shows the validity of the proposed system, especially in real time application aspect, and also shows the feasibility of using a new computer interface instead of the mouse.

Keywords – computer vision, eye feature, gaze tracking, template matching, real time, image-based.

1. Introduction

Eye movement analysis is useful in many applications such as security systems, HCI(human computer interface), visual telephone communication, and systems for man-machine interaction. Also application areas include human factors research and human performance evaluation.

In an extensive study, Yarbus showed that the perception of a complex scene involves a complicated pattern of fixations, where the eye is held still, and saccades, where the eye moves to foveate a new part of the scene. And the eye must be moved in such a way that the target object can be inspected with a higher acuity, by foveating the object. The movement of the eyes to a new location is performed by executing a saccade; yet this is not the only type of eye movement. Research literature classify seven different eye movement; convergence, rolling, saccades, pursuit motion, nystagmus, drift and microsaccades, physiological nystagmus.[1]

Generally, techniques for eye gaze tracking are classified into: a)those that measure the eye position relative to the head, such as the method proposed in the paper and Electroocculargraphy(EOG) approach which is measurement of an electrical potential between the front and the back of the eyeball, b)those that measure the orientation of the eye in space such as cornea reflection method or the video-based method.

Over the past two decades many computer-based systems have been developed for the study of eye movements. It basically consists of two stages, namely, tracking the eye features and analyzing the pattern of eye dynamics. Eye feature tracking usually refers to the estimation of the eye pointing direction, either within the eye socket, or with respect to some scene being viewed.[6]

In this paper, we propose the image-based method that extract the eye features in real time, that track the feature points and that calculate the eye gaze direction. The remainder of the paper is organized as follows. The eye feature extraction is addressed in Section 2. The head motion tracking is constructed in Section 3. Section 4 provides experiment results of eye and head tracking to demonstrate the accuracy aspects.

2. Eye Motion Tracking

In this section, we concisely describe the image preprocessing that is needed to determine the initial position of iris template and the boundary of search area. We describe the device which tracks eye motions. And we explain the proposed method: the iris template method combined with the polygon matching method. The features which inform eye motions are defined, extracted and tracked by using the eye tracking device.

2.1 Preprocessing

In order to determine the initial position of the iris template, images obtained by the camera are processed.
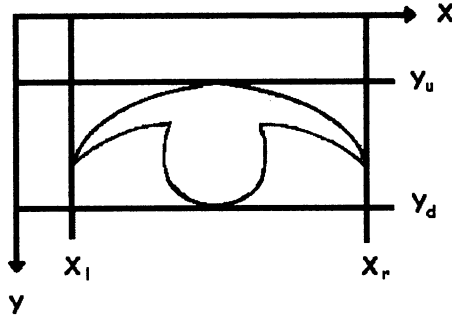
Fig 1. Initial locations of an eye window

This preprocessing of the image informs the locations of an eye window.

We need two initial processes; 1) initializing the location of an eye window to match a polygon to the boundary of an eye and 2) initializing the position of an iris template that represents the boundary of an iris, a black sclera of the eye. Two valley images that have a high gray level in a dark area are obtained. After thresholding, one represents a whole region of eye window like Fig. 1, and the other represents an iris region.

To make segmentation more robust, the threshold value should be automatically selected by the system. Generally, knowledge about the objects in the scene, the application, and the environment should be used in the segmentation algorithm to form more general. This method supposes that in an indoor situation objects ɔupy about P percents of the image area. By using this knowledge to partition the gray value histogram of the input image, one threshold value can be chosen that assigns P percents of the pixels to the object. This is called the p-tile method. Clearly, this method is of very limited use. But in the situation this algorithm makes more robust to the intensity variance.

First, we take four feature points to model an eye boundary. Three of those are located in lower eyelid and the other in upper eyelid. Because of geometrical
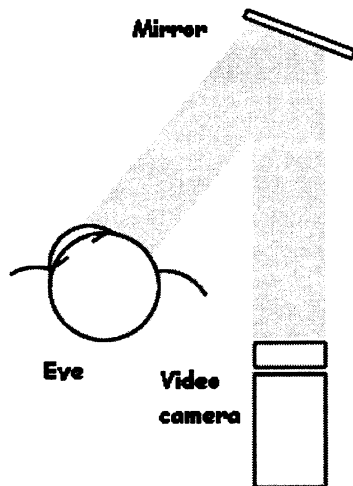

Fig 2. Eye tracking subsystem

constraints in eye motions, the feature that is located in upper eyelid is varying and the rest are not. So we can track the point in real-time. And the locations of three points that are obtained off-line are used.

To set three parameters, radius(r) and position(X_c), initial values of them are determined from the average of valley points that are obtained in the valley image that represents an iris region.

2.2 Description of eye tracking system

The eye tracking system is able to estimate the angular position of the eye which is independent of lateral motion of the video system relative to the head. This is of critical importance since perfect mechanical stabilization of the instrument to the head is impossible during dynamic head movements.

The image of the eye is reflected off of a mirror positioned in front of the subject's eye and directed to the camera, like Fig. 2. The video image is sampled by a custom charge-coupled device(CCD) array that allows images to be sampled at 30Hz. Images from the CCD camera are processed in real time to obtain the estimates of eye boundary, pupil center and radius.

2.3 Template matching

The iris region of the eye image is assumed to be a circle. Thus it is meant to the problem of fitting the image to the template. In order to determine the parameters of the regions, appropriate energy functions are defined using the information about the valleys and edges of the image. The template interacts with the image by adjusting its parameters to minimize (or maximize) the energy function.[3][4]

2.4 Energy function

To effectively extract the iris region, we define the iris template to be a circle. In the template, there are three parameters (X_c, r) corresponding to a circle.

Energy terms are defined in terms of this template to account for the characteristics of the real image of the iris, and the priori relations between the parameters.

The definition of each energy term is given as follows:

1. The energy function, E_1, is used for expanding a radius of the template.

$$E_1 = \sum_{i \in circle} valley\_image_i$$

where $valley\_image_i$ is the value of a binarized valley image and summation is made over the circular area of the template.

2. The energy term, E_2, is a measure of the fitness between the iris template and the valley image.

$$E_2 = \frac{1}{Area_{circle}} \sum_{i \in circle} valley\_image_i$$

3. The term is used for matching a boundary of the template to the edge image.

$$E_3 = \frac{1}{Length_{circle}} \sum_{i \in boundary} edge\_image_i$$

where $edge\_image_i$ is the value of a binarized edge image.

In the above energy functions, valley image which is corresponding to dark region has high potential. And edge image obtained by using morphology filters is a boundary of the binarized valley image. We update the parameters of the iris template such that these energy functions have extreme(minimum or maximum).

3. Head Motion Tracking

The six degrees of freedom (position, orientation) of head movements are measured by a magnetic position sensor. A receiver which is a small (25.4mm×25.4mm×20.3mm) lightly weighted transducer is appended to an eye tracking system. The magnetic position sensor consists of two parts: a transmitter which generates a magnetic field and a receiver which senses position and orientation of itself. This information is transmitted to microprocessor. And the microprocessor processes analog signal of sensing data. Processed data provide 3-dimensional position and orientation(roll, azimuth, elevation) of a head. Measurement quality is influenced by low frequency electro-magnetic interference (EMI) and the presence of large metal objects in the field. To ensure performance, the transmitter and receiver are located away from sources of EMI such as monitors and power supplies.[7]

The head tracker has a translational linear range of ± 91cm and an angular measurement linear range of $\pm 180°$ in azimuth and roll, and $\pm 90°$ in roll. The output power of the head tracking transmitter is adaptively changed in order to maximize signal-to-noise (SNR) ratio without saturation. Within 24cm, adaptive gain control increases transmitter power as the distance between transmitter and receiver increases such that SNR is relatively constant. Outside this range SNR deteriorates as the magnetic field strength decreases with
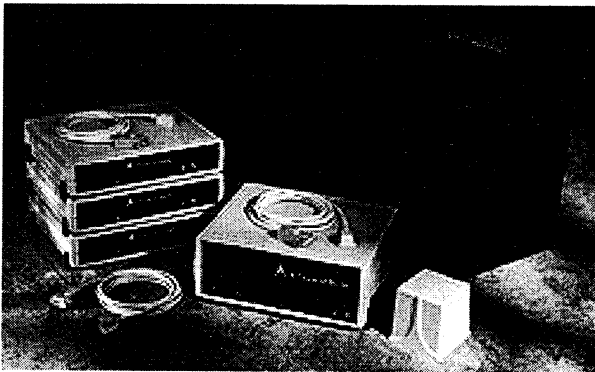

Fig 3. Magnetic position sensor

Angular range:	±180° Azimuth & Roll ±90° Elevation
Static Accuracy Position:	0.07 inch RMS
Static Accuracy Orientation:	0.5° RMS
Static Resolution Position:	0.02 inch @ 12"
Static Resolution Orientation:	0.1° @ 12"
Update rate:	Up to 144 measurements/second

Table 1. Specification of magnetic position sensor

distance. With the receiver within 24cm of the transmitter, the angular resolution of the system is less than $0.1°$ root mean square and the translational resolution is less than 0.75mm (at 48 and 60 cm the azimuth resolution is $0.75°$ and $1.78°$ rms, respectively). The noise spectra of the head tracker is found to be flat throughout the 60-Hz bandwidth at a typical value of $6 \times 10^{-3} deg^2/Hz$ with no significant spectral peaks.[5][7]

3.1 Calibration

In order to measure the eye gaze direction of a user, we need two processes. 1) We should measure the position and orientation of the head with respect to the standard coordinates attached to the monitor and 2)we should measure the position and orientation of the eye gaze direction with respect to the coordinate attached to the head. Finally we can obtain the lay of user's gaze combining two processes.

In order to perform these processes, first of all, the magnetic sensor should be calibrated to know the relative position and orientation between the transmitter and the monitor (Fig. 4). Firstly, the receiver of the magnetic sensor is sequentially located at three corners of the monitor. Then we know vector a, b and c that determine the monitor location with respect to the transmitter of the magnetic sensor. As a result, we can obtain the position and orientation of the receiver with respect to the monitor using these calibration results because the magnetic sensor measure the relative position and orientation between the receiver and the transmitter.
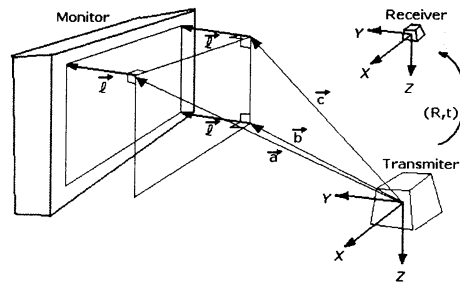

Fig 4.The calibration of the magnetic sensor

4. Experimental result

The proposed algorithm is performed on sequential images using Pentium II 350MHz processor and an image grabber, Meteor board, produced by Matrox. An image grabber continuously provides real images of an eye. And we process the images by using the computer.

4.1 Eye tracking

Images are continuously obtained by using the image grabber, and are displayed 15~20 frames per a second. And the processing time of the proposed algorithm is 50~110ms. Finally we can obtain 6~9 frames of processed images per a second.

Fig. 5 shows the final result of matching eye windows to polygon and of matching an iris template. And Fig. 6 provides the sequence of an iris template matching.

4.2 Head tracking

The error of head motion tracking is the same as resolutions of magnetic position sensor. The error is represented in Table 1.

Head motion tracking is used for HCI. That is to say, the sensor provides the values of the position and orientation and then these values are used for moving a mouse pointer.
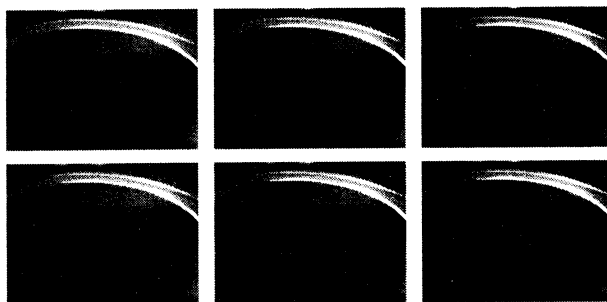

Fig 5. Eye feature extraction

Fig 6. A sequence of an iris template matching
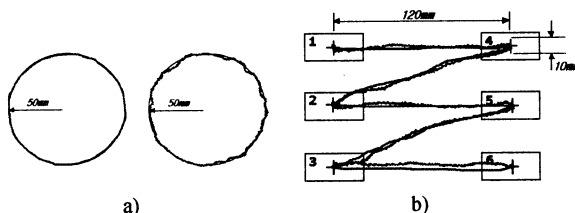

a)   b)

Fig 7. The comparison of the mouse movement by using hand motions and by using head motions; a) a circular movement b) a linear movement

Fig. 7 shows that the difference between the mouse movement by using hands and that by using head motions is small enough to neglect.

5. Conclusion

In this paper we separated the eye gaze tracking into two parts, the eye movement tracking and the head movement tracking. We considered the eye tracking to be independent of head motions. And we tested an interface by using the head-tracking data.

We proposed the method that extracts eye features for the sequential images in real time and showed the experimental results. By simply modeling the eye features we could extract the features in real time.

Eye motions were determined and tracked by the eye features. And head movements were detected by magnetic sensor. Also we showed that the head direction is used for interfacing between the computer and the user.

The further work is combining the eye tracking and the head tracking. The combining system not only extracts and tracks eye and head motions in the same time but also compensates eye motions related with the head motions. Finally, we can expect that an integrated system finds eye gaze direction freely to the head motions

References

[1] A. J. Glenstrup and T. Engell-Nielsen, "Eye controlled media: present and future state", Thesis for Bachelor's Degree, University of Copenhagen, 1995.

[2] K. F. Arrington, "Viewpoint eye tracker", Arrington Research Report, USA, 1997.

[3] X. Xie, R. Sudhakar and H. Zhuang, "On improving eye feature extraction using deformable templates", Pattern Recognition, vol. 27, no. 6, pp. 791-799, 1994

[4] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates", Int. J. Comput. Vision, 8:2, pp. 99-111, 1992

[5] R. S. Allison, M. Eizenman, and B. S. K. Cheung, "Combined head and eye tracking system for dynamic testing of the vestibular system", IEEE Trans. Biomed. Eng., vol. 43, no. 11, November 1996

[6] X. Xie, R. Sudhakar, and H. Zhuang, "Real-time eye feature tracking from a video image sequence using kalman filter", IEEE Trans. SMC, vol. 25, no. 12, 1995.

[7] Flock of Birds: Installation and Operations Guide. Burlington, VT: Ascension Technology, 1991.

[8] J. Merchant, R. Morrissette, and J. I. Porterfield, "Remote measurement of eye direction allowing subject motion offer one cubic foot of space", IEEE Trans. Biomed. Eng., vol. BME-21, no. 4, pp. 115-120, 1981

[9] R. Jain, R. Kasturi, B. G. Schunck, Machine Vision, MIT Press, 1995.

[10] R. C. Gonzalez, R. E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, 1993.

Fuzzy System Modeling based on Subtractive Clustering and Generalized Radial Basis Function Network

Im hyun Jo, Kwee bo Sim, Hoon Kang

School of Electronics and Electrical Engineering. Chung Ang University
221, Heuksuk-Dong, Dongjak-Ku, Seoul, 156-756, Korea

Abstract

In this paper, we construct a fuzzy system model based on the subtractive clustering algorithm and the Generalized Radial Basis Function Network(GRBFN). Using the subtractive clustering algorithm, we represent a system's behavior with a small number of data. We construct a fuzzy system based on it. It is proven that the fuzzy system is equal to GRBFN under some minor restrictions. Therefore we can transform a fuzzy system into GRBFN and train this neural network with backpropagation learning. As a result, we could obtain a more optimized fuzzy system than before. And it is applied to some problems for validation.

Keywords

Fuzzy system, Radial basis function network, Clustering, Neural networks

1. Introduction

Clustering is used in data compression or model construction, like a codebook in Vector Quantization or positioning the center of a rule in Fuzzy logic where it needs representative value of whole data.

In neural network, Competitive Learning, Learning Vector Quantization classify data pattern in on-line. In off-line method, Fuzzy C-Means clustering algorithm is a well known iterative procedure but we are suffering from determining initial values like the number of clusters, a position of centers etc, that have a heavy influence on results.

Mountain method using a grid assume every point can be a center, so we need to set the resolution of a grid. It's simple but computation grows exponentially when dimension of a problem increases. Subtractive clustering algorithm, a simple

modified form of Mountain Method, does not need these initial values that are needed in FCM and only consider not every grid point, but data point, so it can find the center more easily and effectively. We construct fuzzy systems based on these clusters.

A TSK fuzzy model, allowing the crisp value in consequent part that consists of linear function of inputs, can follow a non-linear behavior with small number of rules.

Radial Basis Function Network is based on the biological receptive fields and consists of local receptive field. It trains faster than multilayer neural network does, since only responsive field works in a network. Some researchers showed that functional behavior of Radial Basis Function Network(RBFN) and fuzzy systems are equivalent under the some minor restrictions. Therefore we can easily transform fuzzy system into RBFN, and vice versa.

Generalized Radial Basis Function Network (GRBFN) is equal to fuzzy systems with more smoother limitation than RBFN, because some of the restrictions on RBFN removed. So we use their characteristic more freely.

Backpropagation learning rule is widely used in Neural networks to get a better performance. To reduce error that is defined the difference between a desired output and a real output, BPL update weight value of output layer and hidden layer. A direction of changing weight is a negative gradient of error function.

In this paper, fuzzy system is constructed based on Subtractive clustering algorithm, then transformed into GRBFN. Then the width and the center of RBF is trained by backpropagation method to get a more optimized fuzzy system.

These procedure is applied to function approximation problems for validation.

2. Model Identification

After obtaining cluster centers using Subtractive algorithm, we consider it as a center of fuzzy rule. Given an input vector x, the degree of fulfillment is defined as a function of distance between an input and cluster centers, x_i^*

$$\mu_i = e^{-\|x - x_i^*\|^2} \tag{1}$$

then we compute the output

$$y = \frac{\sum_{i=1}^{c} \mu_i y_i^*}{\sum_{i=1}^{c} \mu_i} \tag{2}$$

Takagi-Sugeno's fuzzy model expresses nonlinear functional behavior with a small number of rules due to a consequence part consists of a linear function of an input.[4]

R^i : if x_i is A_1^i and ... and x_n is A_n^i then
$$y^i = h_i + G_1^i x_i + \cdots + G_n^i x_n$$

They also pointed out that optimizing the parameters in the consequent equations reduces to a linear least-squares estimation problem. To solve these problem, we define

$$\rho_i = \frac{u_i}{\sum_{j=1}^{c} \mu_j} \tag{3}$$

And we can rewrite equation (3) as

$$y = \sum_{i=1}^{c} \rho_i y_i = \sum_{i=1}^{c} \rho_i (G_i x + h_i) \tag{4}$$

$$\begin{bmatrix} y_1^T \\ \cdot \\ \cdot \\ y_n^T \end{bmatrix} = \begin{bmatrix} \rho_{1,1} x_1^T & \rho_{1,1} & \cdots & \cdots & \rho_{c,1} x_c^T & \rho_{c,1} \\ \cdots & \cdots & & & \cdots & \cdots \\ \rho_{1,n} x_n^T & \rho_{1,n} & \cdots & \cdots & \rho_{c,n} x_n^T & \rho_{c,n} \end{bmatrix} \begin{bmatrix} G_1^T \\ h_1^T \\ \cdot \\ \cdot \\ G_c^T \\ h_c^T \end{bmatrix}$$

To be brief, it has the form of AX=B. A is a constant matrix of modified membership values, B is a constant matrix of output values, X is a matrix of parameters that we want to solve. A solution of this equation is obtained by well known pseudo inverse method.

$$X = (A^T A)^{-1} A^T B \tag{5}$$

But, in most of cases we can't find the inverse of A because it's large and becomes singular matrix. So, we must find the answer iteratively. To find out parameters of consequence, X, we use the recursive least squares estimation.

3. GRBFN

RBFN is based on the biological receptive fields and consists of local receptive field. It trains faster than multilayer neural network does since only responsive field works in a network while Multilayer network finds a global response using a sigmoid function.

It consists of two layers. In first layer, the ouput is calculated by distance between the input and the center of basis function, so the input near the center has a high value, and outputs drop to zero as input is far from the center. Roger Jang and Sun showed the equivalence of RBFN and Fuzzy inference system under the some minor restrictions.

The restrictions in the RBFN are

1) The number of receptive field units is equal to the number of fuzzy if-then rules

2) The output of each fuzzy if-then rule is composed of a constant

3) The membership functions within each rule are chosen as Gaussian functions with the same variance

4) The T-norm operator used to compute each rule's firing strength is multiplication

5) Both the RBFN and the fuzzy inference system under consideration use the same method(i,e., either weighted average or weighted sum) to derive their overall outputs.

Typically Gaussian function is chosen as a receptive field unit.

$$R_i(\vec{x}) = \exp\left(-\frac{\|\vec{x} - \vec{c_i}\|^2}{\sigma_i^2}\right)$$

$$i = 1, 2, \ldots, n \qquad (6)$$

$\vec{x}$ is an input vector, $\vec{c_i}$ is a center vector with same dimension as $\vec{x}$, i is the number of radial basis functions. There are two types of outputs. A simpler one is defined as the form of weighted sum , with f_i is a strength of ith receptive field

These restrictions change a little bit in GRBFN. First, GRBFN can take part of an input, not a whole input vector.

$$y = f(\vec{x}) = \sum_{i=1}^{n_0} \theta_i(\vec{x}) \phi_i(\vec{x}) = \vec{\theta}\,\vec{\phi} \qquad (7)$$

Second, the weight is a non-linear function, not a constant, so that we can use Takagi-Sugeno's fuzzy model that expresses consequence part with a linear function of input.

$$y = f(\vec{x}) = \frac{\sum_{i=1}^{n_0} \theta_i(\vec{x}) \phi_i(\vec{x})}{\sum_{i=1}^{n_0} \phi_i(\vec{x})} \qquad (8)$$

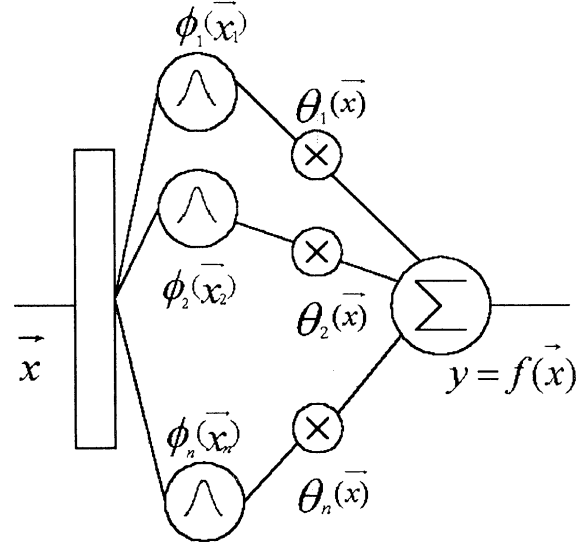Third, it doesn't need to have same variance. The

Fig 2. An GRBFN

shape varies with different width and the Gaussian basis function is defined as

$$\phi_i(\vec{x}) = \exp[-(\vec{x_i} - \vec{c_i})' \Delta_i (\vec{x_i} - \vec{c_i})] \qquad (9)$$

$$\Delta_i = \begin{bmatrix} \frac{1}{\sigma_{i1}^2} & 0 & \cdots & \\ 0 & \frac{1}{\sigma_{i2}^2} & & \\ \vdots & & \ddots & \\ & & & \frac{1}{\sigma_{i(n \times i)}^2} \end{bmatrix}$$

$$(10)$$

To adjust the centers and variance, we train the network with backpropagation method.

Each time we update the values using these equations.

$$c_i(t+1) = c_i(t) - \frac{\partial E}{\partial c_i} \qquad (11)$$

$$= c_i(t) - \eta \frac{\phi_i}{\sum_{i=1}^m \phi_i}(d-y)(w_i - y)\frac{(x_i - c_i)}{\sigma_i^2}$$

$$\Delta_i(t+1) = \frac{\partial E}{\partial \Delta_i} \qquad (12)$$

$$= \Delta_i - \eta \frac{\phi_i}{\sum_{i=1}^m \phi_i}(d-y)(w_i - y)\frac{(x_{ij} - c_{ij})^2}{\sigma_{ij}^3}$$

4. Simulation

To validate these whole process, it is applied to Function Approximation problems.

Convex-shaped function and differential equation are used. The number of training is 100, η_1 and η_2, the ratio of learning is set to 0.1 respectively. Circles in the picture mean a cluster center. Error index shows a system is better optimized after learning.
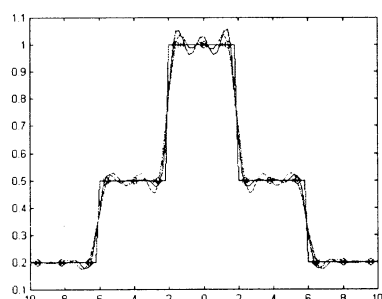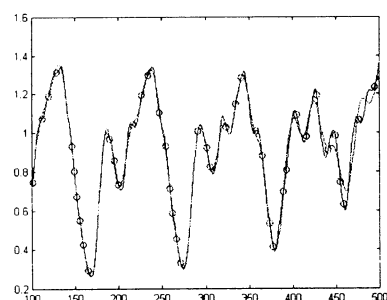

Fig 3. convex-shape function

$$r_a = 0.1$$

Fig 4. Differential Equation

$$r_a = 0.1$$

	Error Index	
	Before learnig	After learning
Convex-shape	0.2521	0.1656
Differential equation	0.0627	0.0531

Table 1. Error Index of function

5. Conclusion

We construct a fuzzy system model based on subtractive clustering algorithm, then we transform it into GRBFN under the restrictions.

We trained this neural network with backpropagation learning and retransform GRBFN into a fuzzy system. After these process we get better results.

It can be combined with Genetic algorithm and other learning techniques.

References

[1] K.J. Hunt, R. H. and R. M.Smith, "Extending the Functional Equivalence of Radial Basis Function Networks and Fuzzy Inference Systems" IEEE Trans. Neural Networks, vol. 7, No. 3, pp. 776-781, 1996

[2] J.S. Roger Jang and C.T. Sun, "Functional Equivalence Between Radial Basis Function Networks and Fuzzy Inference Systems" IEEE Trans. Neural Networks, vol. 4, No. 1, pp. 156-159, 1993

[3] S. L. Chiu "Fuzzy Model Identification Based on Cluster Estimation" Journal of Intelligent and Fuzzy Systems, vol. 2, 267-278, 1994

[4] M.Sugeno and T. Yasukawa, "A Fuzzy-Logic-Based Approach to Qualitative Modeling" IEEE Trans. Fuzzy Systems, vol. 1, No. 1, pp. 7-31, 1993

[5] J.S.R Jang, C.T. Sun and E. Mizutani "Neuro-Fuzzy and Soft Computing" Prentice Hall, 1997

[6] James Bezdek, "Pattern Recognition with Fuzzy Sets and Neural Nets" Second IEEE International Conference on Fuzzy Systems Tutorial #3, 1993

[7] S.A Billinos, G.L Zheng "Radial Basis Function Network Configuration Using Genetic Algorithms", Neural networ, vol. 8, no. 6, pp 877-890, 1995

[8] R. Langari, L. Wang, "Fuzzy models, modular networks, and hybrid learning", Fuzzy Sets and Systems 79, pp. 141-150, 1996

Human Direct Teaching of Industrial Articulated Robot Arms Based on Forceless Control

Daisuke KUSHIDA†, Masatoshi NAKAMURA†, Satoru GOTO† and Nobuhiro KYURA‡

†Department of Advanced Systems Control Engineering, Saga University,

Honjomachi, Saga 840-8502, Japan

‡Department of Electrical Engineering, Kinki University in Kyushu,

11-6 Kayanomori, Iizuka 820-8555, Japan

E-mail: kushida@cntl.ee.saga-u.ac.jp

Abstract

Forceless control means that a robot arm moves appropriately by external force under non-gravity and non-friction condition. In this study, a method of forceless control is proposed for industrial articulated robot arms and is applied for teaching of industrial articulated robot arms. Generally, teaching of industrial articulated robot arms is carried out by use of an operational equipment called teach-pendant. Smooth teaching can be achieved if direct-teaching, in which the robot arm is moved by human's direct force, is applicable. The proposed forceless control enables an easy teaching of the industrial articulated robot arm. The effectiveness of the proposed method was assured by experimental work on a 2 degree freedom articulated robot arm.

Key words Forceless control, Industrial articulated robot arms, Human direct-teaching

1. Introduction

Industrial articulated robot arms are used for many applications of contour control and PTP(Point to Point) control based on teaching-playback method. Then, these operations need the teaching signal to memorize positions of operation. Teaching of industrial articulated robot arms is carried out by use of an operative equipment called teach-pendant. If direct-teaching is possible, teaching is faster. Here, the direct-teaching means that the robot arm is moved by human's direct force to the robot arm by his hand. Actually, direct-teaching is difficult because the friction of joints and the high ratio gears exist and the controllers are already built in. Furthermore, the robot arm is strained by the human hand force during teaching. Non-gravity and non-friction conditions are desirable for the implementation of direct-teaching. Under these conditions, the robot arm will be operated appropriately by the human hand according to the operator's intention. In this research, this kind of the control called "*Forceless Control*", was realized by calculating joint position input according to the ideal motion activated by an external force. By this way, robot arm will move under an external force with no gravity and no friction. In this paper, a method of the forceless control is proposed.
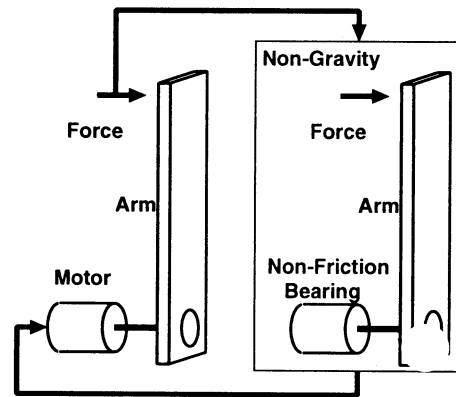

Fig. 1　Conceptual figure of forceless control

Its realization and applications to industrial articulated robot arms are also presented. The realization of direct-teaching is proved by simulation and experimental work on actual industrial articulated robot arms.

2. Forceless Control for Industrial Articulated Robot Arms

2.1 Direct-teaching of industrial articulated robot arms by human hand

Direct-teaching means that robot arm is teaching by human hand directly. Generally, teaching of industrial articulated robot arms was carried out by teach-pendant. But, teaching by teach-pendant is needed human skill because of difference between human's coordinate and robot arm's one. Besides, how to use teach-pendant is different each maker. Hence, direct-teaching is suitable for teaching of industrial articulated robot arms. Besides, teaching time shorten by direct-teaching.

In this study, direct-teaching of industrial articulated robot arms by human hand is realized by using proposed new method called forceless control.

2.2 Conceptual explanation of forceless control

Conceptual figure of forceless control is shown in Fig. 1. Actual robot arm dose not move by external force because it is actuated by the connected motor
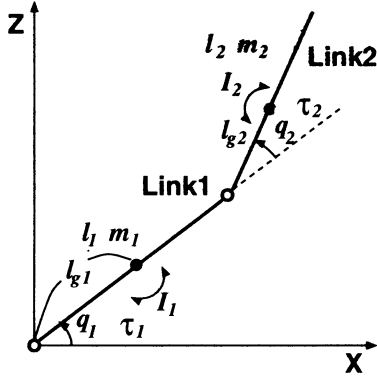
Fig. 2   2 degree of freedom articulated robot arm

through a gear in gravity field. But, if robot arm exists in non-gravity and non-friction condition, it can move under the given external force. The proposed forceless control can realize non-gravity and non-friction motion. Actually, though robot arm is operated by the connected motor, it seems that the robot arm works under the human hand directly instead of motor operation.

2.3 Dynamic equation of industrial articulated robot arms

The dynamic equation of an articulated robot arms is given by

$$H(q)\ddot{q} + D\dot{q} + h(q,\dot{q}) + g(q) = \tau_s \tag{1}$$

where $H(q)$ is the inertia matrix, D is friction matrix, $h(q,\dot{q})$ is the coupling nonlinear terms, $g(q)$ is the gravity term and $\tau_s = [\tau_{s1} \ \tau_{s2}]^T$ is input torque to the robot arm. In case of 2 degree of freedom articulated robot arm as shown in Fig. 2, these parameters are derived as

$$H(q) = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}$$

$$H_{11} = m_1 l_{g1}^2 + m_2(l_1^2 + 2l_1 l_{g2} \cos q_2 + l_{g2}^2) + I_1 + I_2$$

$$H_{12} = m_2 l_1 l_{g2} \cos q_2 + m_2 l_{g2}^2 + I_2$$

$$H_{21} = I_2 + m_2(l_{g2}^2 + l_1 l_{g2} \cos q_2)$$

$$H_{22} = m_2 l_{g2}^2 + I_2$$

$$H(q) = \begin{bmatrix} D_1 & 0 \\ 0 & D2 \end{bmatrix}$$

D_1 :   coefficients of viscous damping of Link1

D_2 :   coefficients of viscous damping of Link2

$$h(q,\dot{q}) = \begin{bmatrix} -m_2 l_1 l_{g2}(2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2) \sin q_2 \\ m_2 l_1 l_{g2} \dot{q}_1 \dot{q}_2 \sin q_2 \end{bmatrix}$$

$$g(q) = \begin{bmatrix} m_1 g l_{g1} \cos q_1 + m_2 g\{l_1 \cos q_1 + l_{g2} \cos(q_1 + q_2)\} \\ m_2 g l_{g2} \cos(q_1 + q_2) \end{bmatrix}$$

where, $q = [q_1 \ q_2]^T$ shows each joint angle. Though input of the dynamic equation (1) is torque, input of actual industrial articulated robot arms is a joint angle because of including control loop. Then, dynamic equation of industrial articulated robot arms must be derived concerning about control loop.
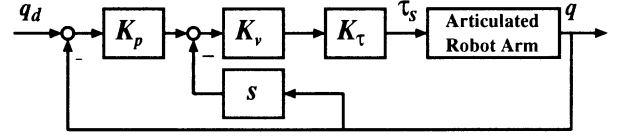

Fig. 3   Control system of industrial articulated robot arm

Control loop of industrial articulated robot arms is shown in Fig. 3 [1], where, K_p, K_v, K_τ are position loop gain, velocity loop gain and torque constant, respectively. The symbols q_d and q are input of joint angle and output of joint angle, respectively. From Fig. 3, the torque input τ_s for a given joint angle input is given by

$$\tau_s = K_\tau[K_v\{K_p(q_d - q) - \dot{q}\}]. \tag{2}$$

Generally, control loop of industrial articulated robot arms is constructed independently. The parameters in the controller are given by

$$K_p = \begin{bmatrix} K_{p1} & 0 \\ 0 & K_{p2} \end{bmatrix}$$

$\quad K_{p1}$: position loop gain of Link1

$\quad K_{p2}$: position loop gain of Link2

$$K_v = \begin{bmatrix} K_{v1} & 0 \\ 0 & K_{v2} \end{bmatrix}$$

$\quad K_{v1}$: velocity loop gain of Link1

$\quad K_{v2}$: velocity loop gain of Link2

$$K_\tau = \begin{bmatrix} K_{\tau 1} & 0 \\ 0 & K_{\tau 2} \end{bmatrix}$$

$\quad K_{\tau 1}$: torque constant of Link1

$\quad K_{\tau 2}$: torque constant of Link2

From the equations (1) and (2), total dynamic equation of industrial articulated robot arms is expressed by

$$H(q)\ddot{q} + D\dot{q} + h(q,\dot{q}) + g(q) = K_\tau[K_v\{K_p(q_d - q) - \dot{q}\}]. \tag{3}$$

2.4 Algorithm of forceless control for industrial articulated robot arms

Flow of forceless control for industrial articulated robot arms is shown in Fig. 4. First, joint torque corresponding to external force must be calculated by use of detected external force by force sensor as follows [2]

$$\tau_f = J^T F \tag{4}$$

where, F is the detected external force, J is Jacobian matrix and τ_f is joint torque corresponding to external force. In non-gravity and non-friction condition, each joint of articulated robot arms are actuated by joint torque τ_f. But, the effect of gravity and friction must be considered. Gravity compensation torque τ_g is obtained by calculating a pose of robot arm in real-time by use of equation (1) as follows
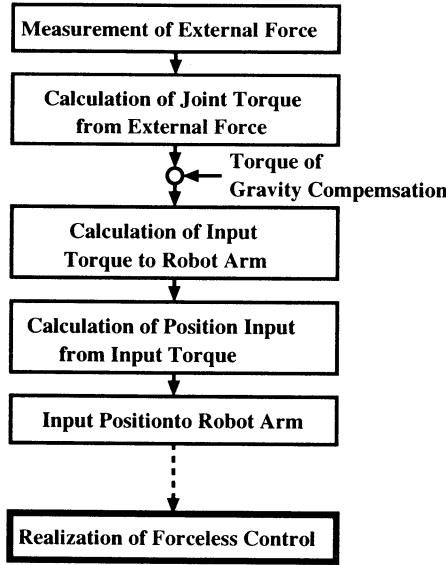
$$\tau_g = g(q). \tag{5}$$

Fig. 4   Flowchart of forceless control

Besides, friction compensation torque τ_d is calculated by obtaining coefficients of viscous damping from actual machine beforehand as followed

$$\tau_d = D\dot{q}. \tag{6}$$

In order to realize forceless control, the input of the joint angle q_d is required as corresponding to these torque by assuming $\tau_f + \tau_g + \tau_d = \tau_s$. By solving equation (2), the input of the joint angle q_d is derived by

$$q_d = K^{-1}\{K_v^{-1}K_\tau^{-1}(\tau_f + \tau_g + \tau_d) + \dot{q}\} + q. \tag{7}$$

Therefore, the forceless control is realized by generating the input joint angle q_d from the external force F. The key point of the method is the realization of the forceless control by inputing appropriate position to control loop without making any change of the control system
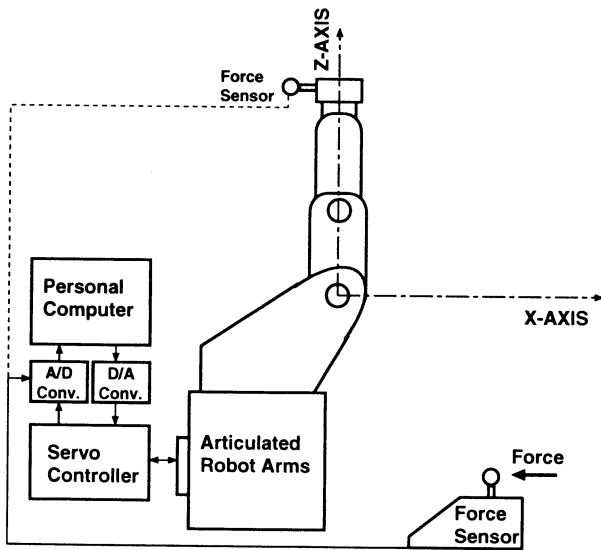

Fig. 5   Construction of experimental equipments

of the robot arm. Besides, robot arm will move by optimum external force because inertia of robot arm can be changed by modifying output of force sensor.

3.   Results of Direct-teaching of Industrial Articulated Robot Arms

3.1   Experimental conditions

The proposed forceless control was applied to a 2 degree of freedom industrial articulated robot arm as shown in Fig. 5. In this research, force sensor was attached at hand for safety operation. External force was detected by force sensor, and input to the personal computer through an A/D converter. In personal computer, objective joint angle was calculated by use of detected external force, and outputting to servo controller through an D/A converter. Besides, joint angular velocity was transfered from servo controller to personal computer. These signal flow was carried out each sampling time. Besides, the used industrial articulated robot arms for experiment was PERFORMER-MK3S(Yahata). Other conditions as follows

(a) Joint angle of Link1
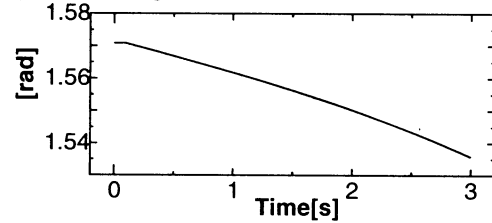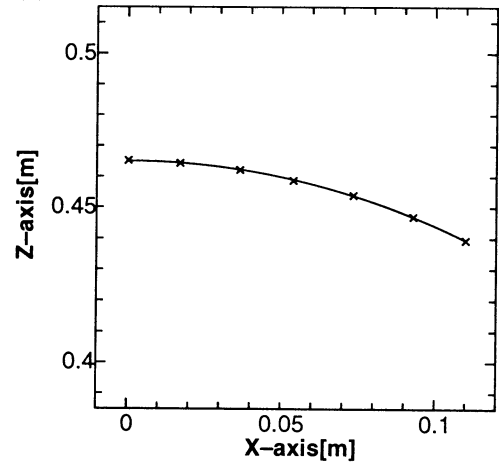

(b) Joint angle of Link2
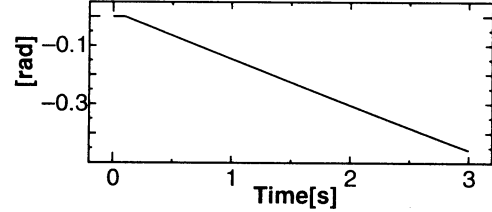
(c) Locus

Fig. 6   Verification of forceless control

- Position loop gain: K_{p1}=25[1/s], K_{p2}=25[1/s]
- Velocity loop gain: K_{v1}=150[1/s], K_{v2}=150[1/s]
- Torque constant:
  $K_{\tau 1}$=0.104[Nms^2/rad], $K_{\tau 2}$=0.061[Nms^2/rad]
- Coefficients of viscous damping:
  D_1 = 4.68[Nms/rad], D_2 = 2.72[Nms/rad]
- Sampling interval: Δt=0.02[s]
- Experimental time: 20[s]

3.2 Simulation results

Before the experiment based on the conditions in the previous section, realization of forceless control must be verified. In this case, initial point of tip of robot arm is $x = 0.0$[m] and $z = 0.465$[m], and tip of robot arm is given 1[N] to the X-direction. Fig. 6 shows simulation results of forceless control of 3 seconds. Fig. 6(a),(b) show trajectory of each joint angle, and (c) shows locus of tip of robot arm. As a result, robot arm was rotated in constant speed by external force. Here, rotation speed is not kept constant because of the interference the arms.

3.3 Experimental results

Experiment of direct-teaching was carried out by using actual industrial articulated robot arms. In this case, teaching was tried without friction compensation during the experiment. Besides, initial position of tip of the robot arm was $x_0 = 0.25$[m] and $z_0 = 0.3$[m], and objective trajectory is the line tends to the bottom vertically. Then, operational velocity in working coordinate was intended almost the same. Experimental results was shown in Fig. 7. Fig. 7(a),(b) show necessary force during direct-teaching, and (c) shows locus of tip of the robot arm. From Fig. 7(c), teaching from initial position to tends to bottom vertically was not so perfectly. Because, coordinate between robot arm and operator is different in the cause of force sensor attached at hand for safety. But it is enough that actual proof which direct-teaching of industrial articulated robot arms will be possible. Besides, direct-teaching was realized by a small force from Fig. 7(a),(b). From this confirmation, effectiveness of forceless control for industrial articulated robot arms was assured, and possibility of direct-teaching of industrial articulated robot arms was shown by experimental work.

4. Conclusion

In this research, forceless control, robot arm will move by external force under non-gravity and non-friction condition, was proposed. Hence, forceless control was tried industrial articulated robot arms including control loop, and applied direct-teaching of industrial articulated robot arms. Verification of forceless control was done under gravity compensation and friction compensation conditions, and effectiveness of forceless control for industrial articulated robot arms was assured by simulation work. Besides, forceless control was applied for direct-teaching of actual industrial articulated

robot arms. Then, friction compensation was removed because of reaction of arm by less force and noise. As a result, possibility of direct-teaching of industrial articulated robot arms was confirmed.
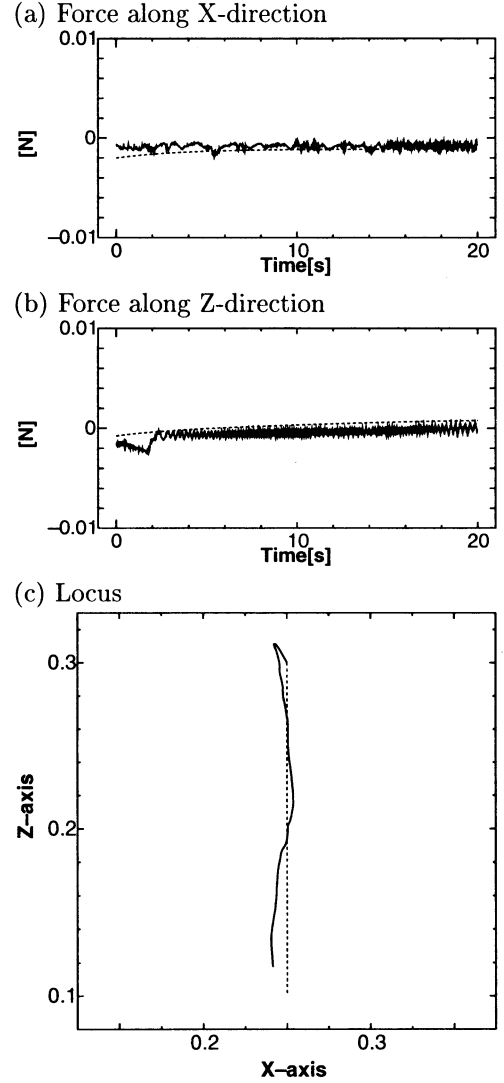
(a) Force along X-direction

(b) Force along Z-direction

(c) Locus

Fig. 7 Direct-teaching based on forceless control(Bold line: experimental results, Dotted line: simulation results)

References

[1] M.Nakamura, S.Goto, N.Kyura(1998), Control of Mechatronic Servo System (in Japanese).

[2] S.Kawamura(1999), Guide of robot control(in Japanese), pp.52-54

Stable Nonlinear Controller Design for Takagi-Sugeno Fuzzy Model

Choon-Young Lee, Tae-Dok Eom and Ju-Jang Lee
Department of Electrical Engineering, KAIST
373-1 Kusong-dong Yusong-gu, Taejon 305-701, Korea
TEL: +82-42-869-5432, FAX: +82-42-869-3410
E-mail: cylee@iliad.kaist.ac.kr, jjlee@ee.kaist.ac.kr

Abstract

This paper proposes another adaptive control scheme for nonlinear systems using Takagi-Sugeno fuzzy model. Takagi-Sugeno fuzzy model has been widely used for identification of structure and parameters of unknown or partially known plants. It shows good approximation capability by fuzzy blending local dynamics. Since Takagi-Sugeno fuzzy model is a nonlinear system in nature and its parameters are not linearly parameterized, it is difficult to design an adaptive controller using conventional design method of adaptive controller. In this paper, local dynamics are assumed to be known, however, its corresponding parameters are unknown. This additional information about system nonlinearity makes it possible to design adaptive controller for nonlinearly-parameterized system. Simulation results have shown that the designed adaptive controller is capable of providing good performance.

1 INTRODUCTION

The primary objective of control problems is to seek some suitable control input which can drive the state of a system to the desired point in the state space. For a system whose dynamics is of complex nonlinear characteristics, the control task becomes difficult due to inexact knowledge about the system and no generalized control methodology so far.

The research on fuzzy modeling, which describes a real system very successfully, has been conducted in the past decade[1] [2] [4]. A stability issue on fuzzy system also has acquired strong attention from many researchers[7][8]. Design methodology for a fuzzy controller has been developed using other nonlinear control theory.

The main advantage of adaptive fuzzy controllers over conventional controller is the use of expert knowledge about the plant dynamics and control strategy. we assumed that we could know its approx-imate description of a nonlinear plant using Takagi-Sugeno fuzzy modeling in advance only around some operating point. Therefore, we know local dynamics about characteristic points and do not have optimal membership functions for that local dynamics. This motivates our research to design adaptive control scheme for a Takagi-Sugeno fuzzy model with gaussian fuzzy membership functions.

A generalized fuzzy adaptive control method was developed by Azam [6] and stable adaptive fuzzy controller for nonlinear systems using supervisory control concept was studied by Li-Xin Wang [3]. This paper used adaptive control scheme for a nonlinearly parameterized system studied by Annaswamy with some improvement on her technique [5]. We derived simple adaptive law using additional information about functional form of system nonlinearity.

This paper is organized as follows. In Section 2, Takagi-Sugeno fuzzy modeling concept is reviewed [1]. In Section 3, an in-depth explanation on the proposed design method is given. In Section 4, simple academic examples are given to illustrate our algorithm. Finally, the conclusion with further research direction is presented in Section 5.

2 BACKGROUND

2.1 Takagi and Sugeno's Fuzzy Model

Takagi and Sugeno [1] proposed an effective way to represent a fuzzy model of a dynamical system. It uses a linear input-output relation as its consequence of individual plant rules. This fuzzy modeling is simple and natural. The main feature of a Takagi-Sugeno fuzzy model is to express the local dynamics of each fuzzy rule by a linear or nonlinear system model. The overall fuzzy model of the system is achieved by fuzzy composition of each local system models. A Takagi-Sugeno model of M rules can be represented as

$$Rule\ i: \quad if\ x_1\ is\ ...\ and\ x_n\ is\ M_n^i \quad (1)$$

$$\text{then} \quad \dot{x} = f_i(x),$$
$$i = 1, 2, ..., r$$

where $x \in R^n$.

For any current state vector x and input vector u, the Takagi-Sugeno fuzzy model infers $\dot{x}$ as the output of the fuzzy model as follows:

$$\dot{x} = \sum_{j=1}^{r} \frac{\mu_j f_j(x)}{\sum_{i=1}^{r} \mu_i} \qquad (2)$$

where

$$\mu_i = \frac{\sum_{j=1}^{n} \phi_j^i e^{-\alpha \phi_j^i}}{\sum_{j=1}^{n} e^{-\alpha \phi_j^i}} \qquad (3)$$

and

$$\phi_j^i = exp\left(-\frac{1}{2}\left(\frac{x_j - c_j^i}{\sigma_j}\right)^2\right) \qquad (4)$$

From (2), it is noted that Takagi and Sugeno's fuzzy model approximates a nonlinear system with a combination of several local systems by decomposing fuzzily the whole input space into several partial spaces and representing each iput-output space with each local dynamic equation.

In our study, we have constructed Takagi-Sugeno fuzzy model as follows:

$$x^{(n)} = \sum_{j=1}^{r} \frac{\mu_j f_j(x)}{\sum_{i=1}^{r} \mu_i} + u \qquad (5)$$
$$= f(\mathbf{x}) + u \qquad (6)$$

where b and functional form of f are known. θ_i are to be updated with adaptation law. The above form is one of Takagi-Sugeno fuzzy model with gaussian membership function. We will investigate the adaptive controller for this type of system model.

3 Stable Adaptive Controller Design

In this section, a stable adaptive controller for nonlinear system is designed by using the information that functional form of nonlinear dynamics is known except its parameters.

Assume that the controlled plant is of the form:

$$x^{(n)} = f(x, \dot{x}, \cdots, x^{(n-1)}, \theta_1, \theta_2, \cdots, \theta_k) + gu \qquad (7)$$

where $\theta_i \in R, i = 1, 2, \cdots, k$ are unknown parameters, u is a scalar control input, f is a scalar function that is nonlinear both in states x and θ_i, g is known. The above plant equation can be rewritten as,

$$\dot{\mathbf{x}} = A\mathbf{x} + b(f(\mathbf{x}, \theta) + gu) \qquad (8)$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \qquad (9)$$

$$b = (0 \ 0 \ 0 \ \cdots \ 0 \ 1)^T \qquad (10)$$

$$\mathbf{x} = (x \ \dot{x} \ \cdots \ x^{(n-1)})^T \qquad (11)$$

The desired reference trajectory x_m is chosen as the output of the model whose dynamics is governed by the differential equation

$$H(s)[x_m] = r \qquad (12)$$

where $H(s)$ is a Hurwitz polynomial and r is a bounded reference input. That is,

$$\dot{\mathbf{x}}_\mathbf{m} = A_m \mathbf{x}_\mathbf{m} + br \qquad (13)$$

The structure of the dynamic system in (7) clearly suggests that when the parameters θ_i are known, a feedback linearizing controller can be found which stabilizes the plant and guarantee output tracking. The state errors between the plant and the model are given by

$$E^{(i)} = x^{(i)} - x_m^{(i)}, \quad i = 0, \cdots, n - 1. \qquad (14)$$

A composite error e_c which is a scalar measure of the state error can be derived as

$$e_c = h^T E \qquad (15)$$

where $h^T(sI - A_m)^{-1}b = \frac{1}{s+k}$.

When system is knonw completely, the control input is then chosen as

$$u = \frac{1}{g}\left(-ke_c - H_1(s)[\mathbf{x}] - f(\mathbf{x}, \theta) + \alpha^T \mathbf{x} + r\right) \qquad (16)$$

where $k > 0$ and $H(s) = s^n + H_1(s)$. Then the error equation can be obtained by simple algebraic manipulation of (7) and (14).

$$\dot{e_c} = -ke_c \qquad (17)$$

If e_c is bounded then E is bounded, and that if $\lim_{t \to \infty} e_c(t) = 0$, then $\lim_{t \to \infty} E(t) = 0$. Therefore, $e_c(t)$ as well as all the state errors tend to zero asymptotically. When θ_i are unknown, the problem is to find a certaintyequivalence controller using (16) and adaptive laws for estimating θ_i so as to achieve global boundedness and asymptotic tracking.

Theorem 1: The following controller and adaptive laws are chosen to result in global boundedness:

$$u = \frac{1}{g}\left(-ke_c - H_1(s)[\mathbf{x}] - f(\mathbf{x},\hat{\theta}) + \hat{\alpha}^T\mathbf{x} + r + u_a\right) \tag{18}$$

$$\begin{aligned} \dot{\hat{\alpha}}_i &= \Gamma_{\alpha_i}e_c\mathbf{x}, \quad \Gamma_{\alpha_i} > 0 \tag{19}\\ \dot{\hat{\theta}}_i &= \Gamma_{\theta_i}e_c\omega_i^*, \quad \Gamma_{\theta_i} > 0 \tag{20} \end{aligned}$$

where ω_i^* is the solution of

$$\omega_i^* = \nabla\mathbf{f}|_{\hat{\theta}_i}, \tag{21}$$

and u_a, additional control input, is the solution of

$$u_a = \text{sign}(e_c)\left(\min\,\text{sign}(e_c)\left[\hat{f} + \Delta\theta\nabla f|_{\hat{\theta}} - \hat{f}(\hat{\theta} + \Delta\theta)\right]\right) \tag{22}$$

where $\theta_i \in \Theta$.

Proof: Let $\tilde{\theta}_i = \hat{\theta}_i - \theta_i$, and $\hat{f} = f(\mathbf{x},\hat{\theta})$ the error dynamics is given as

$$\dot{e}_c = -ke_c + f - \hat{f} + \tilde{\alpha}^T\mathbf{x} + u_a \tag{23}$$

Consider Lyapunov function candidate,

$$V = \frac{1}{2}\left(e_c^2 + \tilde{\alpha}^T\Gamma_\alpha^{-1}\tilde{\alpha} + \tilde{\theta}^T\Gamma_\theta^{-1}\tilde{\theta}\right) \tag{24}$$

Taking time derivative of V along state trajectories,

$$\begin{aligned} \dot{V} &= e_c\dot{e}_c + \tilde{\alpha}^T\Gamma_\alpha^{-1}\dot{\tilde{\alpha}} + \tilde{\theta}^T\Gamma_\theta^{-1}\dot{\tilde{\theta}} \tag{25}\\ &= -ke_c^2 + e_c(f - \hat{f} + \tilde{\alpha}^T\mathbf{x} + u_a) \tag{26}\\ &\quad + \tilde{\alpha}^T\Gamma_\alpha^{-1}\dot{\tilde{\alpha}} + \tilde{\theta}^T\Gamma_\theta^{-1}\dot{\tilde{\theta}} \end{aligned}$$

Applying adaptation law of equation (19) and (20),

$$\dot{V} = -ke_c^2 + e_c\left[f - \hat{f} + \tilde{\theta}\omega + u_a\right] \tag{27}$$

When $e_c > 0$, $\dot{V} \le 0$ if $f - \hat{f} + \tilde{\theta}\omega + u_a < 0$. That is,

$$u_a < (\tilde{f} - \tilde{\theta}\omega) \tag{28}$$

When $e_c < 0$, $\dot{V} \le 0$ if $f - \hat{f} + \tilde{\theta}\omega + u_a > 0$. That is,

$$u_a > (\tilde{f} - \tilde{\theta}\omega) \tag{29}$$

Let

$$\omega^* = \frac{\hat{f} - f}{\hat{\theta} - \theta} = \nabla f|_{\hat{\theta}} \tag{30}$$

Generalizing the equation (28) and (29) into one,

$$u_a = \text{sign}(e_c)\left(\min\,\text{sign}(e_c)\left[\hat{f} + \Delta\theta\nabla f|_{\hat{\theta}} - \hat{f}(\hat{\theta} + \Delta\theta)\right]\right) \tag{31}$$

From the above discussion, (24) is a Lyapunov function. This leads to global boundedness of e_c, and $\tilde{\theta}_i$. From Barbalat's lemma, the above calculation implies that e_c tends to zero.

The proof is complete.

4 SIMULATION RESULT

To illustrate the proposed approach, consider simple academic examples.

Example 1:

Plant model is given by

$$\dot{x} = x^\theta + u \tag{32}$$

where $\theta = 2$.

The reference model is chosen as

$$\dot{x}_m = -x_m + r \tag{33}$$

and reference input is

$$r = sin(t) + sin(2t); \tag{34}$$

Figure 1 shows the response of the plant when proposed adaptive controller is applied. The error converges to zero while adapting plant parameter. Figure 2 shows the estimation of parameter.

Example 2:

Plant model is given by

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{x_1x_2[x_2 + \theta]}{1 + x_1^2 + x_2^2} + u \end{aligned}$$

where $\theta = 2.5$.

The reference model is chosen as

$$\dot{x}_m = A_mx_m + B_mr \tag{35}$$

where $A_m = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}$, $B_m = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

and reference input is

$$r = sin(t) + sin(2t); \tag{36}$$

Figure 3 shows the response of the plant when proposed adaptive controller is applied. The error converges to zero while adapting plant parameter. Figure 4 shows the estimation of parameter.

5 CONCLUSIONS

This paper proposed a design method of adaptive controller of nonlinear systems using Takagi-Sugeno fuzzy modeling and gaussian fuzzy membership function. We assumed that we can model unknown plant around operating point by a Takagi-Sugeno fuzzy model. We designed adaptive controller for Takagi-Sugeno fuzzy model of a nonlinear system using adaptive technique for nonlinearly parameterized system. Although there are some errors between real nonlinear system and fuzzy model, the proposed cotroller can achieve good tracking. Adaptive control problem on a more general nonlinear system using fuzzy model is our further research direction.

References

[1] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 15, pp.116–132, 1985.

[2] J. C. Bezdek, "Editorial: Fuzzy models–What are they and why?," *IEEE Transactions on Fuzzy Systems,* vol. 1, pp.1–6, 1993.

[3] L-X. Wang, "Stable Adaptive Fuzzy Control of Nonlinear Systems," *IEEE Transactions on Fuzzy Systems,* vol. 1, pp.146–155, 1993.

[4] E. Kim, M. Park, S. Ji, and M. Park, "A New Approach to Fuzzy Modeling," *IEEE Transactions on Fuzzy Systems,* vol. 5, pp.328–337, 1997.

[5] A. M. Annaswamy, F. P. Skantze and A.-P. Loh, "Adaptive Control of Continuous Time Systems with Convex/Concave parameterization," *Automatica,* vol. 34, pp.33–49, 1998.

[6] F. Azam and H. F. VanLandingham, "A Generalized Fuzzy Adaptive Control Method," *IEEE Int. Conf. on Systems, Man and Cybernetics,* pp.2083–2088, 1998.

[7] J. Joh, Y. Chen, and R. Langari, "On the Stability Issues of Linear Takagi-Sugeno Fuzzy Models," *IEEE Transactions on Fuzzy Systems,* vol. 6, pp.402–410, 1998.

[8] L. Wang, "Stable and Optimal Fuzzy Control of Linear Systems," *IEEE Transactions on Fuzzy Systems,* vol. 6, pp.137–143, 1998.
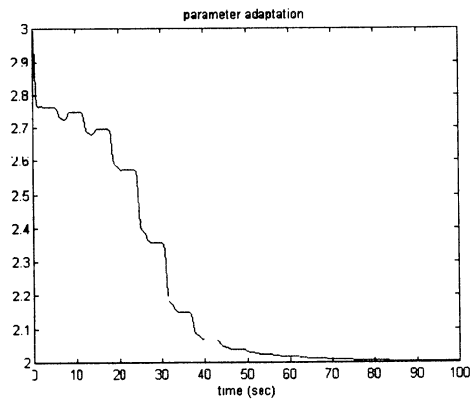
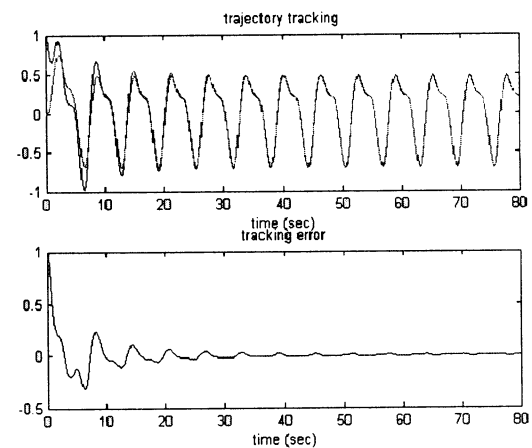Figure 2: parameter adaptation (Example 1)
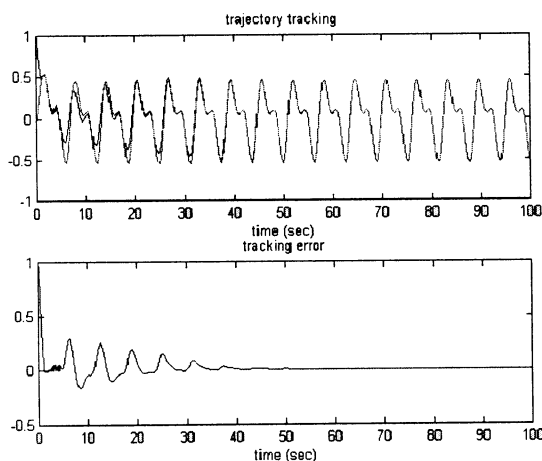

Figure 3: output tracking responses (Example 2)

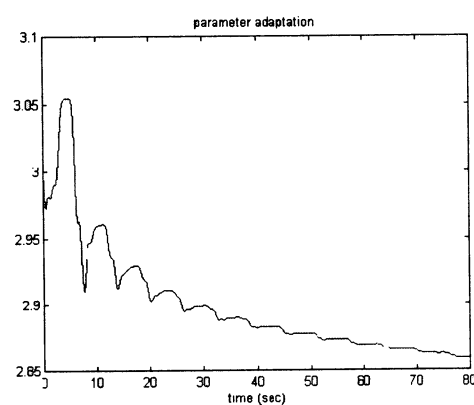Figure 1: output tracking responses (Example 1)

Figure 4: parameter adaptation (Example 2)

Generalized Asymmetrical Bidirectional Associative Memory
for Human Skill Transfer

T. D. Eom
Dept. of Electrical Engineering
KAIST
Taejon, 305-701, Korea

J. J. Lee
Dept. of Electrical Engineering
KAIST
Taejon, 305-701, Korea

Abstract

The essential requirements of neural network for human skill transfer are fast convergence , high storage capacity, and strong noise immunity. Bidirectional associative memory(BAM) suffering from low storage capacity and abundance of spurious memories is rarely used for skill transfer application though it has fast and wide association characteristics for visual data. This paper suggests generalization of classical BAM structure and new learning algorithm which uses supervised learning to guarantee perfect recall starting with correlation matrix. The generalization is validated to accelerate convergence speed, to increase storage capacity, to lessen spurious memories, to enhance noise immunity, and to enable multiple association using simulation work.

1 Introduction

Recently, neural network has found wide applications for modeling and transferring human skills to robot controls. Mainly service robot applications require this type of modeling which includes manipulative skills, locomotions, reaction and a man-machine interface suitable for applications which take place in human centered environment. In designing neural network for human skill transfer, three important aspects of the whole architecture should be considered. First, the stability of overall system should be guaranteed even when the neural network learned by successful human demonstrations fails to control the system due to modeling error. We previously designed a stabilizer which maintains boundness of system trajectory in Lypunov sense [5]. It is a switching scheme which activates robust controller when the system trajectory is trying to deviate over the prescribed threshold. The merit of this switching controller is any sort of neural network can be applied to model human skills. Design of neural network and system stability are divided into independent processes. Second, neural network should be learned fast for on-line applications. Third, neural network should be able to easily approximate arbitrary complex human behaviors by modifying network structure appropriately, *i.e.*, it should be an universal approximator. This paper is mainly devoted to solve the latter two problems by generalizing network sturcture of BAM.

In the early 1980's, Hopfield proposed an auto-associative memory model to store and recall information in much the same way as the human brain. In the late 1980's, Kosko extended the auto-associative memory model to a bidirectional one [1]. This two-level nonlinear memory model is based on earlier studies on associative memories [3]. The bidirectional associative memory (BAM) model is more general and powerful than the Hopfield auto-associative memory and includes the Hopfield memory as a special case. A BAM can associate an input pattern with a different stored output pattern of a stored pattern pair, thus allowing bidirectional association. Owing to their fast learning, good generalization, and noise immunity, BAM's are well suited for modeling human pattern recognition and control behavior.

2 Structure of GABAM

When human recognizes patterns, there are certain feature points more helpful to make a decision. Considering the different usefulness of each pixel information, diagonal matrix $\Lambda^i = diag(\lambda_1^i, \ldots, \lambda_n^i)$ can be multiplied before the correlation matrix. We suggest new forward and backward transition matrix W_f, W_b as follows.

$$W_f = \sum_i \frac{y^i x^{i^T} \Lambda_f^i}{x^{i^T} \Lambda_f^i x^i}, W_b = \sum_i \frac{x^i y^{i^T} \Lambda_b^i}{y^{i^T} \Lambda_b^i y^i}. \quad (1)$$

Applying a pattern x, each y^i is summed with a weighting factor $(x^{i^T} \Lambda^i x)/(x^{i^T} \Lambda^i x^i) = 1 - 2d_{\Lambda^i}(x^i, x)$, if new distance measure, $d_{\Lambda^i}(x^i, x)$, is defined as $(\sum_{x_k \neq x_k^i} \lambda_k^i)/(\sum_k \lambda_k^i)$, where λ_k^i reflects the usefulness of k'th input element. W_b is used in backward association.

3 Learning and Recall Process

Each weighting, $x^{i^T}\Lambda^i x$, is equal to $1 - (x - x^i)^T\Lambda^i(x-x^i)/2$ if each λ_k^i is normalized at every learning step to sum into one. The generalized radial basis function(GRBF) network can be expressed as

$$y = \sum_i w^i f((x - c^i)^T\Lambda^i(x - c^i)), \qquad (2)$$

where activation function $f(u) = e^{-u}$. Though GRBF does not include convergence procedure, GABAM can be translated as GRBF with $w^i = y^i$, $c^i = x^i$, and $f(u) = 1 - u/2$ from the static viewpoint. Original KBAM has no hidden layer. However, augmenting weighting matrix Λ^i derives the virtual hidden layer from the transition matirx. KBAM can rahter be regarded as three-layered network with identity weighting matrix as shown in Fig. 1. The word *virtual* means that it only exists during learning process and merges into single transition matrix in recall process. If the hyperbolic tangent function replaces the usual sign output activation function, whole network is differentiable and any kind of gradient descent algorithm can be utilized to find Λ^i's which minimize the overall classification errors.

For the learning process, backpropagation algorithm which uses only the first derivatives is adopted. To learn the forward association, energy function is denoted by $E = \|y-y^i\|^2/2 = \sum_k(y_k-y_k^i)^2/2$. If stepsize, η, of learning is sufficiently small, this pattern learning has the same performance with batch-type learning for all training pattern pairs. First, feedforward propagation is performed for the current training pair, (x^i, y^i). Calculating the output error, the error is propagated backwards. δ_k^o and δ_i^h denote the backpropagated error in output and hidden layer respectively. Then,

$$\delta_k^o = (y_k - y_k^i), \qquad (3)$$

$$\delta_i^h = \sum_k y_k^i(1 - y_k^2)\delta_k^o, \qquad (4)$$

$$\frac{\partial E}{\partial\lambda_j^i} = -2\frac{m(x_j^i \neq x_j)\sum_k \lambda_k^i - \sum_{x_k^i \neq x_k}\lambda_k^i}{(\sum_k \lambda_k^i)^2}\delta_i^h, \qquad (5)$$

where

$$m(condition) = \begin{cases} 1, & if\ condition\ is\ true, \\ 0, & otherwise. \end{cases} \qquad (6)$$

Finally, λ_j^i is updated by

$$\Delta\lambda_j^i = -\eta\frac{\partial E}{\partial\lambda_j^i}. \qquad (7)$$

The backward assoication is learned in similar way.

4 Properties

The performance of an associative memory is usually evaluated in terms of its storage capacity, noise immunity, and spurious memories. These properties of GABAM were compared with those of the most promising SBAM, ABAM, and GBAM for the pattern recognition problem.

Storage capacities of five different models were compared in case of $m = n = 10$ by randomly generating 1000 test sets for K desired states (x-axis in Fig. 3). GABAM was superior to KBAM, competitive with ABAM and SBAM, and overwhelmed by GBAM. GBAM, which updates all elements in transition matrix, W, and does not contain correlation matrix, is rather similar to feedforward one-layer network. However, GABAM divides W into correlation matrix and input weighting matrix and adjusts only the latter. Therefore, it locates between BAM and feedforward network.

To investigate noise immunity, 26 test pattern pairs(Fig. 2), each consisting of matched small-case and large-case alphabet letters, were considered[2]. Dimensions of input and output patterns were 49(7 by 7 pixels). Fig. 4 shows the recall result of four BAM's in case of random-noise injection from zero to twenty percent. Three variants of GABAM, which used different number of virtual-layer-nodes and training patterns, were suggested. The first variant results in the intermediate performance between GBAM and ABAM. The graphs of the second and GBAM are almost overlapped over all noise injection range because learning algorithm of GBAM, stemming from linear separability condition, trains 49 by 49 weights to properly place hyperplanes, which is up to 49 by 52 weights updated by the second variant of GABAM. Though the linear separability condition limits the capacity of both GBAM and GABAM, GABAM converges faster due to the correlation matrix. The third variant is trained in the broader training set and overwhelms every BAM model.

Percentage of spurious memories was examined by generating 10,000 random initial states and checking whether they converged into spurious states. The results are 71%, 86%, 92%, and 98% for GABAM, GBAM, ABAM, and SBAM respectively(Fig. 5).

5 Multiple Association

In previous human skill transfer approaches, one-to-many mappings in training data are regarded as inconsistency and eliminated by preprocessing. At the current instance, it seems inconsistent but it may be clearly identified considering the previous information. If a single network can discriminate or just display the multiple associations, the network is very useful for the architecture dealing with spatio-temporal sequences.

Symmetric BAM's learn and recall only the fixed pattern pairs through forward and backward association, *i.e.* the "if-and-only-if conditions". Using ABAM many-to-one mapping becomes possible. Multiple association, one-to-many mapping, is actually not a function but a relation. In what situations, do these multiple associations occur ? We assume that it occurs searching inverse mapping with imperfect data. Accessing the database with some keys missing implies the existence of multiple candidates for the matching pattern. In terms of the spatio-temporal sequence, instantaneous multiple association means missing context information. Comparing all the sequences can lead to clarify the correct match.

There are two possible solutions to accomplish the multiple association. One is coding the overlapped characters as new ones. After the convergence the unique parts of multiple patterns can be superposed iteratively. The complexity of implementation gradually increases as the number of multiple association grows. The other way is to include the possibility of imperfectness in model by augmenting the extra dimensions.

If the number of augmented dimension is n_a, 2^{n_a} multiple association can be maximally discriminated. Once each association is assigned the unique identification by values in extra dimensions, all become one-to-one mappings and are easily learned by GABAM which can discriminate highly correlated patterns. This learning procedure can be interpreted as the modeling of exogenous factor. For example, if each double association pair is trained with randomly generated augmented value, it can not be distinguished and becomes ambiguous. However, if synchronized different value is assigned to each association affected by the context, the mapping can be uniquely determined. In the recall process, augmented dimensions are filled with random values before convergence. This results in the exploration of memory through possible links, which is the Markov process depending on the probability of extra values and the connectivity of stored patterns. After convergence, stored pattern cycles are probabilistically generated on the input and output layer. If an extra value is coherent to each pattern and increased by one at every appearance, sequential retrieval can be achieved. Overflow is prevented by modular operation.

Experiments are performed on the alphabetical character set. One example is as follows.
Example :
Case 1 is the double association(Fig. 6). Because only one input pattern is stored, no source of noise from pattern non-orthogonalities leads to the correct recall at one step. One-to-two mapping is converted to double one-to-one mappings augmenting one dimension. After transient pre-convergence sequence(Separated by the first vertical bar in Fig. 6), peroid-4 cycle is repeated using sequential retrieval. The numbers of
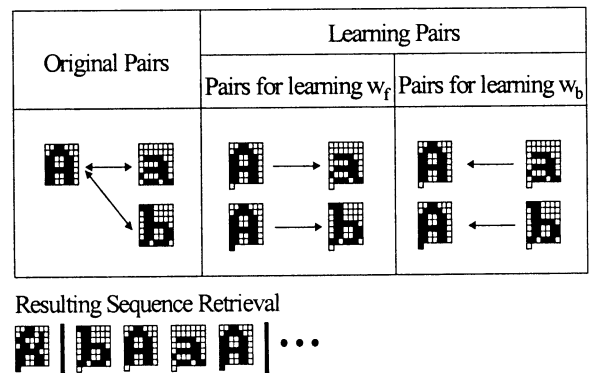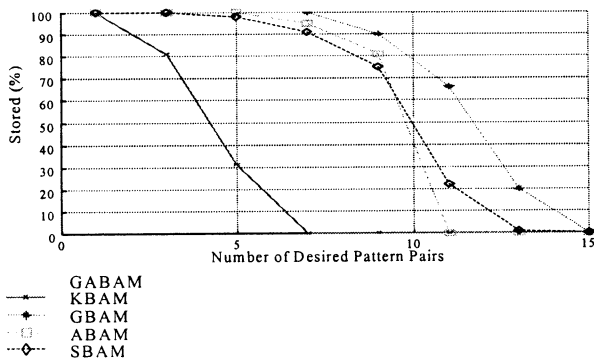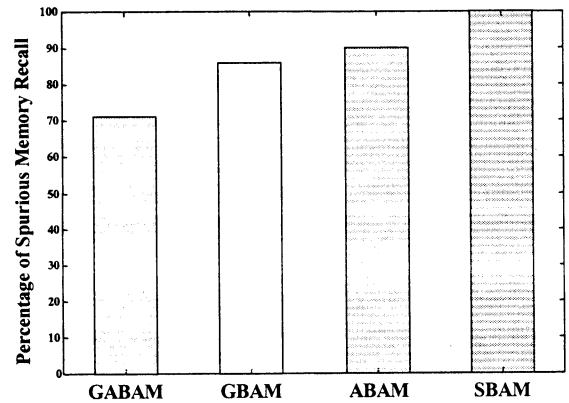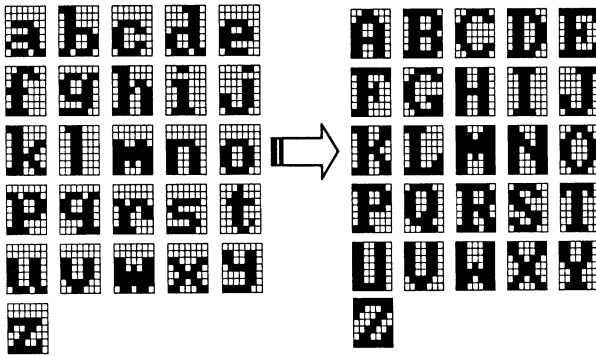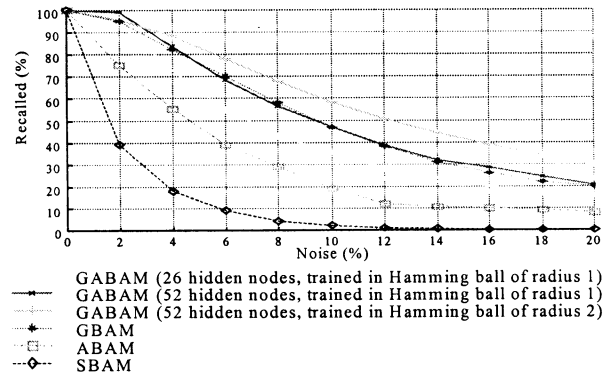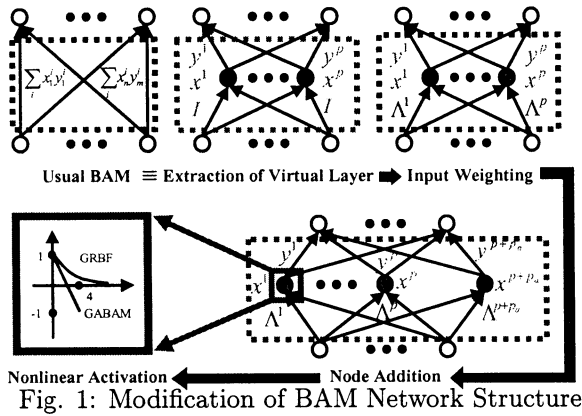
"a" and "b" are equal. It reflects the fact that the numbers of "a" and "b" are same in stored output patterns. Probabilistic retrieval results in the equal possibility of appearance. For the following cases, sequential retrieval is assumed.

6 Conclusion

The concept of BAM was generalized by introducing weighting matrix. The similarity to GRBF enables GABAM to utilize various learning techniques for weighting matrix. The proposed GABAM was proved to have large storage capacity, the best noise immunity, and the least spurious memories among existing models. Also, The capability of storing highly correlated patterns can be efficiently utilzed for multiple association with dimension augmentation scheme modeling missing context. GABAM for multiple assoication aims for learning human control strategy. For the control application, improving network properties through node addition, nonlinear activation, and modified learning algorithm remains as further study

References

[1] B. Kosko, "Adaptive bidirectional associative memories", *IEEE Trans. on SMC*, vol. 18, no. 1, pp. 49–69, 1988.

[2] H. Shi, Y. Zhao, and X. Zhuang, 'A general model for bidirectional associative memories" *IEEE Trans. on SMC*, vol. 28, no. 4, pp. 511–519, 1998.

[3] Y. Hirai, "A Model of human associative processor (HASP)," *IEEE Trans. SMC*, vol. 13, no. 5, pp. 851-857, 1983.

[4] H. Oh and S. Kothari, "Adaptation of the relaxation method for learning in bidirectional associative memory," *IEEE Trans. Neural Networks*, vol. 5, pp. 573-583, July 1994.

[5] T. Eom, M. Sugisaka, and J. Lee, "New skill learning paradigm using various kinds of neurons," *Applied Mathematics and Computation*, vol. 91, no. 1, pp. 9-22, 1998.

Fig. 1: Modification of BAM Network Structure

GABAM (26 hidden nodes, trained in Hamming ball of radius 1)
GABAM (52 hidden nodes, trained in Hamming ball of radius 1)
GABAM (52 hidden nodes, trained in Hamming ball of radius 2)
GBAM
ABAM
SBAM

Fig. 4: Comparison of Models (Noise Immunity)

Fig. 2: Desired Pattern Pairs
for Noise Immunity Test

Fig. 5: Comparison of Models (Spurious Memory)

GABAM
KBAM
GBAM
ABAM
SBAM

Fig. 3: Comparison of Models (Storage Capacity)

Resulting Sequence Retrieval

Fig. 6: Multiple Association (Example)

A New PID Regulation Controller with Lyapunov Stability

Jung-Hoon Lee

Member of ReCapt, Dept. of Control and Instrumentation. Eng.
Gyeongsang National University, 900 Gazwa-Dong, Chinju 660-701, Korea
FAX:+82-591-757-3974, Phone:+82-591-751-5368, email:jhleew@nongae.gsnu.ac.kr

Abstract

In this paper, the stability of second order uncertain systems with PID regulation controllers is analyzed by using Lyapunov second method for the first time in the time domain. By means of the results of this stability analysis, the maximum bound of the error from the nominal output is determined as a function of the gains of the PID control. The robust gain design rule is suggested so that this error can be guaranteed by the prescribed specifications. The usefulness of the proposed algorithm is verified through an illustrative example.

Keywords: PID control, Lyapunov stability, regulation control

I. Introduction

The PID control is the useful and most popular to engineers in almost industrial because of fast response, feedback type, zero steady state error, robust and excellent performance for slow dynamic plants, practical proof, and trust from field engineers[1]. The reason of the difficulty of tuning the PID control is the different output from the designed output due to the modeling errors, undesired zeros, and the function of the reference value.

Since Ziegler and Nichols proposed the design methods of the PID gains in 1942[2], the many methods has been suggested[1,3,4-17], for example, model based pole assignment[4-7], optimal method[6], bang-bang+PI[8], method for integral wind-up[9,10], phase and gain margin satisfying PI[11,12], and auto-tuning automation[13,14,18,17] etc. The most studies on the design of PID controller are concentrated on the selection or tuning of the suitable and stable PID gains. However, the works about the time domain stability analysis due to disturbance and parameter variations and the robust gain selection are not reported except only fact that the zero steady state error.

In this paper, the stability of 2nd order servo regulation systems such as a position control of dynamic plants investigated using the Lyapunov second method for the first time in time domain. The gain design rules to guarantee the specifications on the error from the nominal output is suggested. A MATLAB simulation with a BLDC direct drive motor is given to show the usefulness of the proposed design rules.

II. Lyapunov Stability Analysis
A. Plant Description

In the position control of dynamic plants such as motors, plants can be modeled as a second order system

$$\dot{X}_1 = X_2$$
$$\dot{X}_2 = -a_1(t) \cdot X_1 - a_2(t) \cdot X_2 + b(t) \cdot U - F(t) \qquad (1)$$

where X_2 is the speed, X_1 is the position, U is the PID control, $F(t)$ is the disturbance, and $a(t)$ and $b(t)$ are the uncertain system parameters expressed as

$$a(t) = a_1^0 + \Delta a_1(t), \quad a(t) = b_2^0 + \Delta a_2(t), \quad b(t) = b_1^0 - \Delta b_1(t) \quad (2)$$

where a_1^0, a_2^0, and b^0 are the nominal values. The $\Delta a_1(t)$, $\Delta a_2(t)$, $\Delta b(t)$, and $F(t)$ are known bounded as

$$\Delta a_{1M} = \max\{|\Delta a_1(t)|\} \qquad \Delta a_{2M} = \max\{|\Delta a_2(t)|\}$$
$$\Delta b_M = \max\{|\Delta b(t)|\} \qquad F_M = \max\{|F(t)|\} \qquad (3)$$

and without any loss of generality, $b(t)$ is assumed to be bounded positive and its min. and max. values represented as

$$0 < b_m < b(t) < b_M, \quad b_m = \min\{b(t)\}, \quad b_M = \max\{b(t)\} \quad (4)$$

B. PID Control

For regulating the output of the plants (1) to the reference X_R, the typical type of the PID control becomes

$$U = K_P \cdot e(t) + K_I \cdot \int_0^t e(\tau)d\tau + K_D \cdot \dot{e}(t) \qquad (5)$$

where the error $e(t)$ and its derivative $\dot{e}(t)$ are defined respectively as

$$e(t) \equiv X_R - X_1 \text{ and } \dot{e}(t) \equiv -\dot{X}_1 = -X_2. \qquad (6)$$

The steady state value of the integral in the PID control can be obtained as

$$I_S = \int_0^{t_S} (X_{1R} - X_1)d\tau = \{a_1(t_s)X_R + F(t_s)\}/[b(t_s)K_I] \qquad (7)$$

where t_s implies the time in the steady state. As a result of the PID control, the closed loop servo regulation system expressed in moving average model becomes

$$\dot{X}_1 = X_2$$
$$\dot{X}_2 = -(a_2^0 + b^0 K_D) \cdot X_2 - (a_1^0 + b^0 K_P) \cdot X_1 - b^0 K_I \int_0^t X_1 d\tau$$

$$+b^0 K_P \cdot X_R + b^0 K_I \int_0^t X_R d\tau] - D(t) \qquad (8)$$

where $D(t)$ is the lumped disturbance as

$$D(t) = \Delta a_1 X_1 + \Delta a_2 X_2 + F + \Delta b U \qquad (9)$$

Since the lumped disturbances clearly influence on the output of the servo system from (8), many engineers notice the robustness problems for all the lumped disturbances. In this paper, this robustness problems can be easily solved by means of the analysis about the stability of the PID regulation control servo system with Lyapunov second method in time domain.

C. Analysis of Lyapunov Stability

To analyze the stability of the PID regulation servo system, first of all, a Lyapunov candidate function is taken as

$$V = 1/2 X^T \cdot X, \quad X^T = [e_0 \ e \ \dot{e}] \qquad (10)$$

where e_0 is defined as an error of the integral to the steady state value of the integral, i.e.:

$$e_0 = I_s - \int_0^t e(\tau) d\tau \qquad (11)$$

Differentiating (10) with respect to time and rearranging it with (7) and (8) leads to

$$\dot{V} = e_0 \cdot \dot{e}_0 + e \cdot \dot{e} + \dot{e} \cdot \ddot{e}$$

$$= -e_0 \cdot e + e \cdot \dot{e} + \dot{e} \cdot [-a_1^0 e - a_2^0 \dot{e} - b^0 U] + (a_1^0 X_R + D)\dot{e}$$

$$= -e_0 e - (a_1^0 - 1 + b^0 K_P)e\dot{e} - (a_2^0 + b^0 K_D)\dot{e}^2 + b^0 K_I e_0 \dot{e}$$

$$\qquad - b^0 K_I I_s \dot{e} + (a_1^0 X_R + D)\dot{e}$$

$$= -X^T Q X + D' \qquad (12)$$

where D' is defined as

$$D' = \{a_1^0 - a(t_s)b^0 / b(t_s)\} X_R \dot{e} + \Delta a_1 X_1 \dot{e} + \Delta a_2 X_2 \dot{e} + \{F(t)$$

$$\quad - F(t_s)b^0 / b(t_s)\} \dot{e} + \Delta b K_P e\dot{e} + \Delta b K_I \int_0^t e(\tau) d\tau \dot{e} + \Delta b K_D \dot{e}^2 \qquad (13)$$

and a positive definite matrix Q should be non singular, unfortunately not unique for example, Q are

$$Q = \begin{bmatrix} 0 & 2 & -2b^0 K_I \\ -1 & 0 & 2(a_1^0 - 1 + b^0 K_P) \\ b^0 K_I & -(a_1^0 - 1 + b^0 K_P) & (a_2^0 + b^0 K_D) \end{bmatrix} . \qquad (14)$$

The choice is givn to desigenrs, good rules will be discussed. Finally, the following equation can be obtained

$$\dot{V} < 0 \quad \text{for} \ \| X \| > X_B \qquad (15)$$

$$X_B = (\| D' \| / \| Q_i \|)^{1/2} \qquad (16)$$

where X_B is a stable bound for PID regulation servo systems as shown in Fig. 2. Therefore the PID control exhibits the bounded stability in the sense of Lyapunov with respect to the uncertainties and disturbances. Hence after estimating the stable bound, the error from the nominal output can be estimated as functions of the gains K_P, K_I, and K_D, further more, it is possible to derive the rule to determine the gains of PID control so that the maximum error form the nominal output is guaranteed by a finite desired value.

D. Robust Gain Design Rule

From (3), (4), and (16), the estimated stable bound becomes

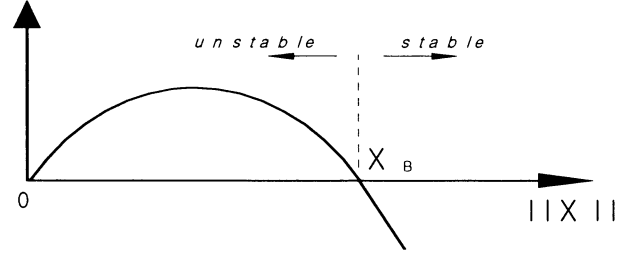$$\hat{X}_B = (\| D'' \| / \| Q_i \|)^{1/2} \qquad (17)$$

Fig. 1 Stable region in sense of Lyapunov

$$D'' = \left|\{a_1^0 - a(t_s)b^0 / b_m\}\right| X_R \| \dot{e} \| + \Delta a_{1M} \| X_1 \| \dot{e} \|$$

$$+ \Delta a_{2M} \| X_2 \| \dot{e} \| + \left|\{F(t) - F(t_s)b^0 / b_m\}\right| \dot{e} \| + \Delta b_M K_P \| e\dot{e} \|$$

$$+ \Delta b_M K_I \left|\int_0^t e(\tau)d\tau\right| \dot{e} \| + \Delta b_M K_D \| \dot{e}^2 \| \qquad (18)$$

In (18), X_R, X_1, X_2, e, $\dot{e}$, and $\int_0^t edt$ are physicall bounded,

for example, generally $\max.|e| = X_R$ if PID control systems are stable. Let e_a be the specification of the allowed maximum error from the nominal output. The gains of the PID control will be designed in the two steps, i.e., stable design and robust design in order to satisfy the speification on the maximum error from the nominal output in the presence of the uncertainties and disturbances. i.e. $e_a > \hat{X}_B$.

D.1 Stable Design Rule

The gains of PID control should be designed for Q to be positive real, that is all the eigenvalue of Q are positive. The characteristic eqn. of Q becomes

$$|\lambda - Q| = \lambda^3 - (a_2^0 + b^0 K_D)\lambda^2 + 2\{(a_1^0 - 1 + b^0 K_P)^2$$

$$+ (b^0 K_I)^2 + 1\}\lambda - 2(b^0 K_I)(a_1^0 - 1 + b^0 K_P)$$

$$- 2(a_2^0 + b^0 K_D) = 0 \qquad (19)$$

The each condition for the stability can be derived for the positive definite matrices of Q as

$$\therefore \begin{cases} K_D > -(a_2^0 / b^0) \\ (a_1^0 - 1 + b^0 K_P)^2 > -(b^0 K_I)^2 - 1 \\ (b^0 K_I)(a_1^0 - 1 + b^0 K_P) > -(a_2^0 + b^0 K_D) \end{cases} \qquad (20)$$

If (20) is satisfied, the PID control system can be stable. Thus, using the condition of (20), the gains of the PID control will be used in view of the stability as a minimum requirement of general control systems.

D.2 Robust Design Rule

The robust design rule will be discussed through the two cases, i.e, with only disturbances and with uncertainties and disturbances.

i) With uncertainties and disturbances

The following relationship of the error specification and the estimated stable bound should be satisfied because the maximum error from the nominal output is resulted from the overshoot due to the damping condition of the gains and uncertainties and disturbance itself
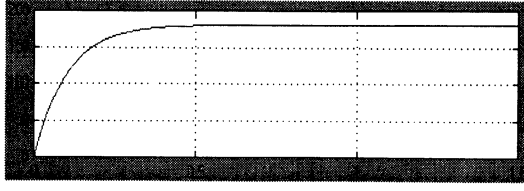
(a) Output response without load variations
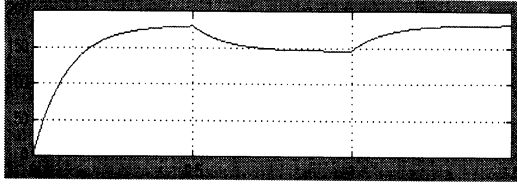
(b) Output response with load variations

Fig. 2 Simulation results for case i) 20[%] error($e_a = 36°$)

$$\therefore \begin{cases} K_P > \{-(a_1^0 -1)e_a + \Delta a_{1M}(|X_{R1}| + |X_1|) + \Delta a_{2M} \\ \quad + F_M + \Delta b_M K_I |\int_0^t edτ| + \Delta b_M K_D |\dot{e}|\}/\{(b^0 - \Delta b_M)e_a\} \\ e_a > \max\{os_\Delta(\xi)\} + \max\{\hat{X}_B\} \end{cases}$$ (21)

where $os_\Delta(\xi)$ means the maximum overshoot according to the damping condition of the uncertain system with the gains K_P, K_I, and K_D due to the variations of the uncertainties and disturbances.

ii) With only disturbances

$$\therefore \begin{cases} K_P > \dfrac{\{F_M / e_a - a_1^0 + 1\}}{b^0} \\ e_a > os_n(\xi) + (F_M \cdot |\dot{e}| / \|Q_i\|)^{1/2} \\ \geq os_n(\xi) + (\|\{F(t) - F(t_s)b^0 / b_m\}|\ddot{e}\| / \|Q_i\|)^{1/2} \end{cases}$$ (22)

where $os_n(\xi)$ means the overshoot according to the damping condition of the nominal system with the gains K_P, K_I, and K_D due to the variations of the disturbances, if critical and over damping conditions, then $os(\xi) = 0$. Therefore, the PID gain to satisfy the specification on the error from the output without variations of the uncertainties and disturbances can be determined using the stable design and robust design rule. There may be an iterative design in order to satisfy both design rules. Then this design method to determine the gain of the PID control can guarantee the maximum error from the output without the variation of the uncertainties and disturbances. In addition to, it can be applicable to analyze the induced error due to changes of disturbance and uncertainties for the previously chosen gains.

III. Simulation Studies

To show the usefulness of the suggested gain design rules, the simulation on the position control of a BL direct drive servo motor as shown in Table 1 will be carried out under only load disturbances. Using the parameters in Table 1, a nominal model of the motor can be obtained as

$\dot{X}_1 = X_2,$

$\dot{X}_2 = -54.25 \cdot X_2 + 12446 \cdot U$ (23)
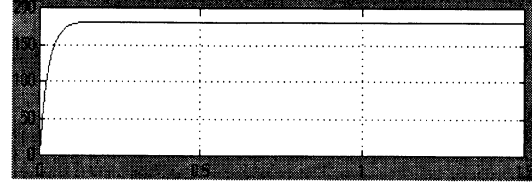
(a) Output response without load variations
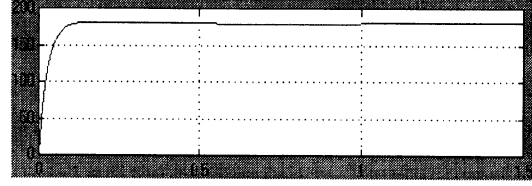
(b) Output response with load variations

Fig. 3 Simulation results for case ii) 1[%] error($e_a = 1.8°$)

A reference command is given as 180° typically. Full load didturbance is applied from 0.5[sec] to 1.0[sec]. Under this condition, for the two cases of the error from the nominal output, i.e, i) 20[%] of command($e_a = 36°$) and ii) 1[%] of command($e_a = 1.8°$), the PID gains are designed by the stable design rule, (20) and the robust design rule, (22) equation for matrix Q as the representative and the no overshoot condition as an inherent rule

case i) 20 [%] error($e_a = 36°$)

.condition .gain selection .pole/zero analysis

$$\begin{cases} K_P > 1.99 \\ K_I > 0.0 \\ K_D > 0.0044 \end{cases} \rightarrow \therefore \begin{cases} K_P = 2.0 \\ K_I = 0.1 \\ K_D = 0.2 \end{cases} \rightarrow \therefore \begin{cases} s_z = -0.5 \\ s_{p1} = -0.1 \\ s_{p2} = -9.8 \\ s_{p3} = -2533.6 \end{cases}$$ (24)

case ii) 1 [%] error($e_a = 1.8°$)

.condition .gain selection .pole/zero analysis

$$\begin{cases} K_P > 39.79 \\ K_I > 0.0 \\ K_D > 0.0044 \end{cases} \rightarrow \therefore \begin{cases} K_P = 40 \\ K_I = 10 \\ K_D = 1 \end{cases} \rightarrow \therefore \begin{cases} s_z = -0.25 \\ s_{p1} = -0.252 \\ s_{p2} = -39.7 \\ s_{p3} = -12460. \end{cases}$$ (25)

where $s_{pi}, i = 1, 2, and\ 3$ are the poles of the closed loop transfer function, one of them is relatively close to the zero, and $\lambda_i, i = 1, 2, and\ 3$ are the eigenvalues of the matrix Q. As a results of the computer simulations on the position control of the motor by the designed PID, Fig. 2 and 3 show the no overshoot output responses of the two cases without the load variations in (a) and with the load variations in (b). In Fig.2, 35.1° (19.5[%]) error from the nominal output at 1⁻ [sec] and -35.04° (19.47[%]) error

Item	Value	Unit
Rated Power	120	[Watt]
Rated Torque	11.0	[Nm]
Rated Speed	123	[rpm]
Rated Voltage	70.0	[Volt]
Rotor Inertia	0.00156	[Kgm²]
Current Constant	3.038	[Nm/A]
Number of Pole	16	

Table 1 Characteristics of a motor

from the output without load variations after the first disturbance at 0.5[sec] at 1.5[sec] appear. In Fig. 3, 1.73° (0.96[%]) error from the nominal output at 0.6[sec] and −1.7° (0.94[%]) error from the output without load variations after the first variation at 0.5[sec] at 1.1[sec] can be found. These results show satisfaction of the given specification on the error as designed. If the output is designed with overshoot, the overshoot due to the variations of the disturbance, $os_n(\xi)$ that can be found with the nominal plant and maximum value of the load disturbance is considered in the design of the controller.

As can be seen from now, the forcus of the this paper is concentrated on the analysis of the effect of the disturbance variations and guaranteeing the error specifications from the output without the vairations, not on the shape of the output that is . Generally, the larger gain, the smaller error due to the same disturbances. The technique in this paper can provide the accurate values of the gains how large gains should be selected.

IV. Conclusions

In this paper, the stability of second order servo systems regulated by PID type controllers is analyzed by using Lyapunov second method for the first time in the time domain. The stability property of PID regulation servo systems is discovered, that is the bounded stability in sense of Lyapunov. By means of the results of this stability analysis, the maximum bound of the error from the output without variations of the disturbances and uncertainties is determined as a function of the gains of the PID control, which can be applied to the effect analysis for the output response due to the variations of the disturbances and uncertainties. And using the relationship of the PID gain and maximum bound of the disturbances and uncertainties, the rules for selection of the PID gain, named as the stable and robust design rules, are suggested so that the error from the output without the variations of the disturbances and uncertainties can be guaranteed by the prescribed bound. Although those rules are not unique and analytic, it is accurate and useful to find the gain solution how large values are sufficient to satisfy the given error specifications. The usefulness of the proposed algorithm is verified through an illustrative example with the MATLAB simulations of a position regulation servo control of a brushless direct drive motor by the PID algorithm with the suggested gains. Therefore, the gain design techniques for PID controls in the paper can give rise to the capability of analysis of the effect of variations of the disturbance to the output response, i.e, calculation of the error from the output without variations and possibility of gain design to satisfy the specification on the error given by users.

References

[1] K. J. Astrom and T. Hagglund, Automatic tuning of PID controllers.. Instrument society of America, 1988.

[2] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Trans. ASME*, Vol.64, pp.759-768, 1942.

[3] R.E. Kalman and J. E. Bertram, "Control system analysis and design via the "second method" of Lyapunov," *Trans. ASME* vol. 82 no.3 pp. 371-393, 1960.

[4] K. J. Chien, J. A. Hrones and J. B. Reswick, "On self tuning regulators, *Trans ASME*, Vol. 74, p.175, 1952.

[5] S. Fukuta, A. Mohri, and M. Takata, "On the determination of the optimal feedack gains for multivariable linear systems incorporating integral action," *Int. J. Control*, Vol. 31, no. 6, pp.1027-1040, 1980.

[6] S. Marsili-Libelli, "Optimal design of PID controller," *Int. J. Control*, Vol.33, no.4, pp.601-616, 1981.

[7] T. Hagglund and K. J. Astrom, "Automatic tuning PID controller based on dominant pole design," *Proceedings of the IFAC Conference on Adaptive Controle of Chemical Processes*, Frenfurt, 1985.

[8] I. H. Suh, S. H. Hwang, Z. N. Bien, "A design and experiment of speed controller with PI-plus bang-bang action for a DC Servomotor with transistorized PWM drives," *IEEE Tran. Ind. Electronics*, Vol, IE-31, no.4, pp.338-345, 1984.

[9] N. J. Krikelis and S. K. Barkas, "Design of tracking systems subject to actuator saturation and integral wind-up," *Int. J. Control*, Vol.39, no.4, pp.667-682, 1984.

[10] Youbin Peng, Damir Vrancic, and Raymond Hanus, "Anti-Winuo, Bumpless, and Conditioned Transfer Techniques for PID Controllers," *IEEE Control Systems*, pp.48-57, 1996..

[11] K. J. Astrom and T. Hagglund, "Automatic tuning of simple regulators with specfications on phase and amplitude margins," *Automatica*, Vol. 20, pp.645-651, 1984.

[12] W. K. Ho, C. C. Hang, and L S. Cao, "Tuning of PID controllers based on Gain and Phase Margin Specifications," *Automatica*, Vol. 31, pp.497-502, 1995.

[13] K. J. Astrom and C. C. Hang, P. Persson, and W. K. Ho, "Toward intelligent PID control," *Automatica*, Vol. 28, ni.1 pp.1-9, 1992.

[14] Y. Nishikawa, N. sannomiya, T. Ohta, and H. Tanaka, "A method for auto-tuning of PID control parameters," *Automatica*, Vol. 20, no. 3, pp.321-332, 1984.[

[15] R. Ortega and R. Kelly, "PID self-tunners: Some Theoretical and practical aspects," *IEEE Tran. on Ind. Electronics,* Vol.IE-31, no.4, pp.332-337, 1984.

[16] C. L. Nachtigal, "Adaptive controller performance evaluation: Foxboro EXACT and ASEA Novatune," *Proceedings ACC-86*, pp.1428-1433, 1986.

[17] Z. Y. Zhao, M. Tomizuka and S. Isaka, " Fuzzy gain scheduling of PID controllers," *IEEE Trans. Systems, Man & Cybernetics*, vol.23, no5, pp.1392-1398, 1993.

Primary control strategy based on the idea of Abrain

Masanori Sugisaka*, Xin Wang**

*Department of Electrical and Electronic Engineering, Oita University,
700 Dannoharu, Oita 870-1192 Japan
Fax: +81-547841, Tel: +81-975-547831, E-mail: msugi@cc.oita-u.ac.jp
**Department of Electronic Engineering, Oita Institute of Technology,
27-407 Higasihama, Nakatsu 871-0006 Japan
Fax: +81-97-5524953, Tel: +81-90-8836-4099, E-mail: wang@oita-it.ac.jp

Keywords: NEUC, On-line learning, Control

Abstract

The research on designing a general structure of artificial brain (Abrain) has been described in [1]. In this paper, we introduce how to put the idea of Abrain into the application controlling a mobile vehicle. The main system of the mobile vehicle consist of a neurocomputer, a Von Neumann computer (NC) and a camera using charge-coupled device (CCD). The hardware neurocomputer RN-2000 (NEUC) is the kernel part of the system. The purpose of this paper is to solve the problem how to realize the hardware neurocomputer by back propagation (BP) neural network learning online. The strategy presented in this paper is based on modifying the past patterns and adjusting the content of the driving patterns by a new algorithm. Learning happens during the driving procedure of the mobile vehicle. This research shows the possibility of the neurocomputer which the BP neural network is inside can learn human knowledge on-line.

1 Introduction

Generally, the configuration[1] of an artificial brain should consists of

various sensors which receive different information, a self-organizing unit, several parallel meurocomputers and Von Neumann computer, a unit for generating internal performance criterion, and the decision units which determines control actions to drivers or actuators.

The items above are the foundation of our researches.

More details about CCD camera sensor have been illustrated in [2], only some words are given here. The environmental information after imaging processing is concentrated into an image plane. The image of an object is divided into three parts, concerning centers of gravity are caught in each part respectively. This method of image processing is good to use more image information.

The hardware NEUC is the kernel part of the system. The purpose of this paper is to solve the problem how to realize the hardware neurocomputer by back propagation (BP) neural network learning on-line. The strategy

presented in this paper is based on modifying the past patterns and adjusting the content of the driving patterns by a new algorithm. Learning happens during the driving procedure of the mobile vehicle. This research shows the possibility of the neurocomputer which the BP neural network is inside can learn human knowledge on-line by the aid of software. Some words about our past reasearhes are also given to help understanding the new method.

2 Problems to be solved

The on-line control of the mobile vehicle is based on the neural control and applied the new algorithm into the teaching patterns. The key point of this new algorithm is utilize a cost function to evaluate the teaching patterns used in the NEUC, and then produces new teaching patterns. These new patterns will be used to train the neurocomputer again. This algorithm needs several driving periods till satisfactory driving results are practiced. The control system is shown in Fig.1 and the classification of driving patterns are illustrated in Fig.2. In normal case, the former patterns are used to train the mobile vehicle. If a teaching pattern changes nothing, what the mobile vehicle has learnt will be no change. In order to get more driving knowledge, pattern changes are hopeful. Let us introduce a cost function $J(t)$. The cost function are definited as the difference between the driving position $q(t)$ and desired position $q_d(t)$ at the time t, the cost function is also used to operate the logical switch $K1$ and $K2$ to control which patterns are put into use.

Let J_{max} be the maximum allowing driving difference, if at a time index t, driving difference $J(t) = \|q(t)\text{-}q_d(t)\| <= J_{max}$, switch $K1$ will be in state off and $K2$ in state on, former patterns are not be changed. Otherwise $K1$ will be in the state on and $K2$ in the state off, former patterns will be put into a renewing and reproduction process. To the NEUC, the teaching patterns must be kept in the form of memo note document, so the results of pattern changes must be represented in one of the memo note document and consist of head mark, pattern number, numbers of input patterns and output patterns, input-teaching patterns and

Table 1: Teaching patterns document

Content	Notes	Name
RNC-DAT	head mark	/
7	total patterns	/
1	BPN' number	/
7 1	I/O neurons	/
127 0 0 0 0 0 0	input pattern	pattern1
10	output pattern	pattern1
0 127 0 0 0 0 0	input pattern	pattern2
20	output pattern	pattern2
0 0 127 0 0 0 0	input pattern	pattern3
30	output pattern	pattern3
0 0 0 127 0 0 0	input pattern	pattern4
40	output pattern	pattern4
0 0 0 0 127 0 0	input pattern	pattern5
50	output pattern	pattern5
0 0 0 0 0 127 0	input pattern	pattern6
60	output pattern	pattern6
0 0 0 0 0 0 127	input pattern	pattern7
70	output pattern	pattern7

output-teaching patterns. During a driving procedure, the driving result is detected by the cost function $J(t)$, and teaching patterns are adjusted at the same time, the new algorithm reproduces patterns according to the basic patterns which cause of $J(t) \leq J_{max}$, and deleted the patterns which made $J(t) > J_{max}$. And the total patterns are limited in 64 because of the construct of the neurocomputer hardware.

An example of real training patterns in a neural controller is presented in Table 1.

3 How the neurocomputer works

The RN-2000 neurocomputer (NEUC) can be connected with a Von Neumman computer (NC) by a bus. In this case, it is operated by software RN-2001 installed in the computer. The NEUC use the same inputs/outputs (I/O) interface with the. The I/O addresses is up to the NC. Data are processed in the NC's memory. If the NECU is connected with not only NC but also some other devices, the input/output data can be from/to either NC or these devices.

The NEUC required special formats of teaching patterns. The input neuron can only accept one of two values 0 or 127 and the output value should not be larger than 127. That means those inputs to the input neurons must be a value of 0 or 127. So the data after image processing have to be changed in order to form 0 or 127 input patterns.

The NEUC which is based on the back propagation algorithm is used in the control system. The NEUC consists of seven RN-200 digital neural network VLSI chips and sixteen neurons in each layer, and whole 256

synapses are integrated in a $13.73 \times 13.73 mm^2$ VLSI chip. This chip can perform 5.12 giga pulse operations per second. It corresponds to effective neural computing rate of 40M CPS.

The problems are how to design a training pattern, how to select the layers of the neural network, how to decide the neuron in each and how those inputs after image processing are transferred into the neurons in the input layer.

The decision of the parameters of the neural network in the NEUC. It is necessary to select the parameters of the neural network in the NEUC. These parameters are the numbers of input neurons, the numbers of output neurons, the middle layers in the neural network, the neurons in each middle layer, learning rate of the BP used in the NEUC[4]. Nobody can say clearly what is the best relationships among layers and neurons. The NEUC has four layers, one input layer, two middle layers and an output layer. Though that available neurons in each layer can be up to 16, and input patterns are 64 are designed. By our experiences on the NEUC, we limited the input neurons in 7, output neuron in 1. The neurons in two middle layers are 16 respectively, this selection is for the purpose of utilizing the capability of the hardware as large as possible. The possible learning rate, which is used to adjust the weights in the neural network, is from 0.1 to 1.0. Based on the experiments, we found no big difference when the learning rate was changed, so 1.0 was selected. The initial weights of the neural network in the NEUC is set randomly.

4 On-line learning results

From the teaching patterns shown in Table 1, it can be seen that the input patterns consist of two strings 127 and 0, the driving knowledge is composed by the different arrangement of 16 possible units (called input neurons) and an output unit (called an output neuron). The new algorithm will change the structure of the neural network in the neurocomputer and the control outputs.

The principle is illustrated in Fig.2. Driving patterns are classified into m classes and every class means a kind of path, for example line, curve and so on. The number of patterns in each class may be equal or not equal. In the case of Fig.2, two patterns are included in *Class* 1, $j - i$ patterns are in *Class k*, and *class m* has only *pattern n*. *Class* 1 present direct path , *Class k* present urgent curve path and *class m* present curve path. Let the $\min\{J11, J12, J13, J14\} = J1_{min}$, $\min\{Jm1, Jm2, Jm3, Jm4\} = Jm_{min}$. If the $J1_{min} = J12$, and $Jm_{min} = Jm1$, respectively, the patterns are formed based on the cost function J12, and Jm1, in *Class* 1 and *Class m*. Here Jl_{min} $(l = 1, 2, \cdots, m)$ is the minimum cost function in the four cost function in a same driving procedure. The on-line learning needs a

Table 2: Teaching patterns document

Content	Notes	Name
RNC-DAT	head mark	/
7	total patterns	/
1	BPN' number	/
13 1	I/O neurons	/
127 0 0 0 127 0 0	input pattern	pattern1
14	output pattern	pattern1
127 0 0 0 0 127 0	input pattern	pattern2
16	output pattern	pattern2
127 0 0 0 0 0 127	input pattern	pattern3
18	output pattern	pattern3
127 0 0 0 0 0	input pattern	pattern4
15	output pattern	pattern4
0 0 127 0 0 0 0	input pattern	pattern5
30	output pattern	pattern5
0 0 0 127 0 0 0	input pattern	pattern6
40	output pattern	pattern6
0 0 0 0 127 0 0	input pattern	pattern7
50	output pattern	pattern7
0 0 0 0 0 127 0	input pattern	pattern8
60	output pattern	pattern8
0 0 0 0 127 0 127	input pattern	pattern9
56	output pattern	pattern9
0 0 0 0 0 127 127	input pattern	patternA
63	output pattern	patternA
0 0 0 0 0 0 127	input pattern	patternB
70	output pattern	patternB
127 0 0 0 0 127 127	input pattern	patternC
77	output pattern	patternC
127 0 0 0 127 0 127	input pattern	patternD
84	output pattern	patternD

certain distance to begin because at least 2 cost function are necessary in a class. There are two cases:

Case 1: If $Jl_{min}>J_{max}$, this is considered that the teaching signal is not good enough and caused a driving difference which does not be allowed. In order to produce more skillful driving knowledge, the past patterns had to be reformed. In the control system of the mobile vehicle, the output-teaching signal is the steering control invariable to the motor driver. If the output-teaching signal is adjusted according to the $Jl_{min}(t)$, then the good propelling results can be expected and after several driving procedure, the x in Eq.(1) to get $Jl_{min}(x) < J_{max}$ is possible. We use the class 1 in Fig.2 as an example to illustrated the process. Assume that J_{max}=200mm, $J1_{min}(t)$=min {J11(t), J12(t)}=J12(t) =min{400,300}=300mm at a time index t, and the former pattern 2 is corresponding to the pattern 2 in Table 1 and the output-teaching signal is 20. That 10 and 20 in *class* 1 caused $\|J1_{min}(t) - J_{max}\|$=100 driving difference. So the 10 is regard as bad one, four new patterns are formed according to the pattern 2. In Table 2

the output teaching patterns are calculated by: pattern 4: (20+10)/2=15, pattern 3: 20×90%=18, pattern 2: 20×80%=16, pattern 1: 20×70%=14, Finally, the data is wrote into the former memo note and start the new learning process. If necessary, the class can be divided into more detail as shown in Fig.2.

Case 2: If $Jl_{min}<J_{max}$, it is regarded as good teaching patterns, then the all of teaching patterns will be renewed according to this good pattern. We take the pattern 7 in Table 1 as example. Assume that J_{max}=200 mm, and at a time index t $J7_{min}(t)$=J71(t) =90mm (here, m=7). Table 2 is a complete teaching document.

From Table 2 we can see that the new patterns more details classification than the human knowledge, and the knowledge increase gradually. The new patterns in Table 2 is used to train the neurocomputer. Compared the results of the teaching patterns in Table1 and Table 2, new patterns contribute obvious good features to the mobile vehicle control system.

5 Conclusions

Teaching signal is necessary for BP neural network to learn human knowledge. In normal ways, these teaching signals have to be made. In the mobile vehicle control system, a neurocomputer, which BP neural network is inside, is applied not only to control the driving action, but also realize the learning on-line. In order to get better results, learning on-line is one of important method, though it is difficult. The strategy shown in this paper pays attention into those output-teaching patterns and utilize a new algorithm to realizes the combinatorial near-optimization method. This idea not only suitable to improve the learning way of the neuro-computer RN-2000, but also helpful to this kind of BP neural network.

Although a CCD camera sensor is availabe way to catch information into the prime Abrain, it is considering for more than one CCD cameras to do better and more works. Supersonic device is a good kind of device to range finding. Moving objects can be detected by light sensors. All of the information can be handled in some micro processors and these processors are connected with one or two host computers.

6 References

[1] Sugisaka, M(1997) "Neural networks for control in an artificial brain of recognition and tracking system", *Proc. of Int. Symposium on Artificial Life and Robotics*, Beppu, Oita Japan, Feb.18-20, pp. 234-237
[2] Sugisaka, M. Wang X(1998) "Artificial brain for a mobile vehicle", *Proc. of Int. Symposium on Artificial Life and Robotics*, Beppu, Oita Japan, Jan.19-21, pp. 275-278

Task Priority Control of Redundant Manipulators

Youngjin Choi
Graduate Student
Dept. of Mech. Eng.
yjchoi@postech.ac.kr
POSTECH

Wan Kyun Chung
Associate Professor
Dept. of Mech. Eng.
wkchung@postech.ac.kr
POSTECH

Sang Rok Oh
Principal Research Scientist
Div. of Elec. and Info. Tech.
sroh@amadeus.kist.re.kr
KIST

Abstract

In the redundant manipulators, the subtask can be specified in terms of redundancy and a sequence of tasks can be simultaneously implemented using task priority algorithms. Now, we propose a new resolution method, "Augmented Jacobian Method" for task priority resolution. The performance of this algorithm is investigated in terms of two important criteria, algorithmic singularity and task error, through the comparative study for existing algorithms.

1 Introduction

The concept of task–priority algorithm for manipulators with redundancy was introduced by Nakamura et al[3]. According to the order of priority, the task with higher priority is firstly performed and the task with lower priority is performed utilizing the redundancy. However, the conventional schemes suffer from kinematic and algorithmic singularities. The kinematic singularity has been the intrinsic characteristics for robot manipulators, on the contrary, the algorithmic singularity is the artificial one which can be eliminated by amending the algorithm.

We will investigate advantages/disadvantages of existing algorithms(Nakamura's and Chiaverini's algorithm) in detail, and make clear the characteristics of two algorithms. Ultimately, we derive the augmented Jacobian approach from the comparative study and propose AJM(Augmented Jacobian Method) which overcomes algorithmic singularity without affecting the primary task. Also, it minimizes the error of secondary task. For future notations, $\mathcal{R}(*)$ and $\mathcal{N}(*)$ represent the range space and null space of matrix $*$, respectively.

2 Comparative Study

To begin with, consider Nakamura's algorithm for task priority redundancy resolution scheme proposed in [3]. In general, the task is specified by *forward differential kinematics* between the task velocity $\dot{p} \in \Re^m$ and joint velocity $\dot{q} \in \Re^n (n > m)$ as follows:

$$\dot{p} = J(q)\dot{q}, \tag{1}$$

(a)Nakamura (6) (b)Chiaverini (10)
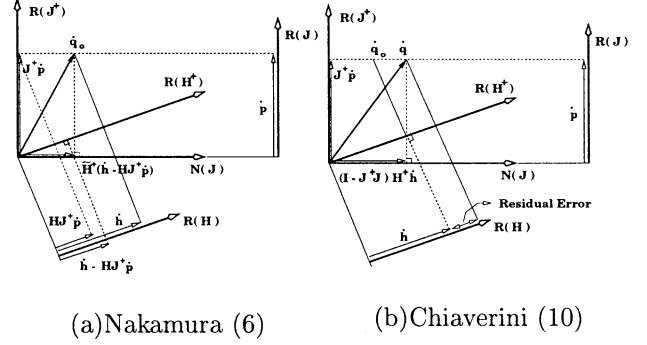
Figure 1: Comparison of Existing Algorithms

where the Jacobian matrix $J(q) \in \Re^{m \times n}$ is obtained by the kinematic structure of the manipulator. Redundant manipulator has the kinematic redundancy in the sense that there still remains $r = n - m$ dimensional set of self motion velocities which does not interfere with the task velocity. In general, the primary task defined by (1) can be implemented as following form

$$\dot{q} = J^+\dot{p} + \left\{ I - J^+J \right\} z, \tag{2}$$

where the matrix J^+ is the pseudoinverse and z is the arbitrary vector. In Nakamura's algorithm, the vector z should be chosen to fulfill some secondary requirements. The secondary task is also specified by the differential form of

$$\dot{h} = H(q)\dot{q} \in \Re^l, \tag{3}$$

where $H(q) \in \Re^{l \times n}$ is Jacobian matrix for the secondary task with lower priority. If we are to achieve the minimum in the sense of the least square error of

$$\|\dot{h} - H\dot{q}\|_2 = \|\dot{h} - HJ^+\dot{p} - H\left\{ I - J^+J \right\} z\|_2, \tag{4}$$

then the least square solution z is obtained by

$$z = [H\left\{ I - J^+J \right\}]^+ \left\{ \dot{h} - HJ^+\dot{p} \right\}. \tag{5}$$

If $H\left\{ I - J^+J \right\}$ has full row rank less than r(the dimension of redundancy), the least square solution is not unique and contains an additional homogeneous term. Allowing the duplicate use of notation, it results in

$$\begin{aligned} \dot{q} &= J^+\dot{p} + \widetilde{H}^+ \left\{ \dot{h} - HJ^+\dot{p} \right\} \\ &+ \left\{ I - J^+J \right\} \left\{ I - \widetilde{H}^+\widetilde{H} \right\} z \end{aligned} \tag{6}$$

where

$$\widetilde{H} = H \left\{ I - J^+ J \right\} \in \Re^{l \times n}. \qquad (7)$$

The residual error becomes zero if $\dot{p} \in \mathcal{R}(J)$ and $\widetilde{H}$ retains full row rank, that is $l \leq r$ and $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) = \emptyset$. As shown in Figure 1.(a),[1] if $\mathcal{R}(H^T)$ does not overlap with $\mathcal{R}(J^T)$, then the scheme of (6) is well performed and does not occur the algorithmic singularity. However, it is always possible for the range spaces of $\mathcal{R}(J^T)$ and $\mathcal{R}(H^T)$ to overlap for an instant. Then this causes the algorithmic singularity. Basically, there are two kinds of singularities in solving *inverse kinematics*, the one is the kinematic singularity and the other is the algorithmic singularity. The kinematic singularity takes place in the case of either

$$rank(J) < m \quad \text{or} \quad rank(H) < l \qquad (8)$$

where the latter case is also called "the secondary task singularity" and the algorithmic singularity occurs in either

$$\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) \neq \emptyset \qquad \text{when } l \leq r$$
or
$$\mathcal{N}(H) \bigcap \mathcal{N}(J) \neq \emptyset. \qquad \text{when } l > r, \qquad (9)$$

where $l < r$ means that the secondary task is under-specified and $l > r$ over-specified for the degrees of redundancy. The kinematic singularity is the fundamental problem in solving the inverse kinematics, but the algorithmic singularity is an artificial part different from the kinematic singularity. Algorithmic singularity can be eliminated or changed according to the characteristics of algorithm. Irrespective of the class of singularities, it should be noted that the nominal algorithm of (6) does not produce an acceptable solution near singularities. Although the damped least square method has been utilized to recover the continuity of inverse kinematic solution, the error of primary and secondary tasks remains in Nakamura's algorithm.

To overcome the algorithmic singularity problem, Chiaverini[1] modified the task priority scheme of (6) as following form

$$\dot{q} = J^+ \dot{p} + \left\{ I - J^+ J \right\} H^+ \dot{h} \\ + \left\{ I - J^+ J \right\} \left\{ I - H^+ H \right\} z. \qquad (10)$$

Although the scheme of (10) does not include algorithmic singularities, we can easily see that the error of secondary task is generally nonzero as shown in Figure 1.(b). The amount of residual error in Figure 1.(b) is calculated by

$$\dot{h} - H\dot{q} = (I - HH^+)\dot{h} - HJ^+(\dot{p} - JH^+\dot{h}), \qquad (11)$$

---
[1]Note that $\mathcal{R}(J^T) = \mathcal{R}(J^+)$ and $\mathcal{R}(H^T) = \mathcal{R}(H^+)$

and the residual error (11) has a strong relation with the algebraic condition between J^+ and H. For example, the secondary task has always the error except that $HJ^+ = 0$, i.e. $\mathcal{R}(H^T) \perp \mathcal{R}(J^T)$.

REMARK 2.1 *(Nakamura[3] vs. Chiaverini[1])* *Although Nakamura's algorithm of (6) has algorithmic singularities, the primary and secondary tasks have no error in normal case which singularities do not occur. On the contrary, Chiaverini's algorithm of (10) has no algorithmic singularities, however, it has nearly always large error for the secondary task.*

3 Augmented Jacobian Approach

Now, we are to eliminate the residual error of (11) shown in Figure 1.(b) by taking a suitable weighted pseudoinverse instead of pseudoinverse. The method of (10) can be modified to

$$\dot{q} = J_W^+ \dot{p} + \left\{ I - J_W^+ J \right\} H^+ \dot{h} \\ + \left\{ I - J_W^+ J \right\} \left\{ I - H^+ H \right\} z, \qquad (12)$$

and the residual error of (12) is given by

$$\dot{h} - H\dot{q} = (I - HH^+)\dot{h} - HJ_W^+(\dot{p} - JH^+\dot{h}), \qquad (13)$$

where $J_W^+ = W^+ J^T (JW^+ J^T)^+$. The algebraic condition to make the residual error of (13) zero is that

$$HJ_W^+ = 0 \rightarrow HW^+ J^T = 0 \qquad (14)$$

for H such that $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) = \emptyset$ with full row rank H and J. As a matter of fact, the condition (14) means the inverse weighted orthogonality. The following lemma can guide the choice of the weight matrix satisfying (14).

LEMMA 3.1 *(Magnus & Neudecker : 1988 [2])* *Let $W = J^T J + H^T H$, then the following statements are equivalent:*

1. $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) = \emptyset$

2. $rank(W) = rank(J) + rank(H)$

3. $JW^+ J^T = JJ^+$

4. $HW^+ H^T = HH^+$

5. $JW^+ H^T = 0$.

By Lemma 3.1, we can see that the weight matrix given by

$$W = J_{AUG}^T J_{AUG} = \begin{bmatrix} J \\ H \end{bmatrix}^T \begin{bmatrix} J \\ H \end{bmatrix} \qquad (15)$$

Figure 2: Augmented Jacobian Scheme of (16)

can satisfy the inverse weighted orthogonality of (14) if the overlapped space between $\mathcal{R}(J^T)$ and $\mathcal{R}(H^T)$ does not exist, i.e. $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) = \emptyset$. Also, the condition of $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) = \emptyset$ implies that the augmented Jacobian J_{AUG} has full rank. In general, the weight is a positive semidefinite matrix if J_{AUG} is under-specified, that is $n > m + l$. The new scheme of (12) has no algorithmic singularity if J_{AUG} retains full rank. However, it is not assured that J_{AUG} retains full rank, hence Lemma 3.1 is not always applicable to new algorithm (12).

Proposed scheme of (12) can be generalized as following form

$$\dot{q} = J_W^+ \dot{p} + \left\{ I - J_W^+ J \right\} W^+ H^T \dot{h} + \left\{ I - W^+ W \right\} z, \tag{16}$$

where the left terms were replaced with right ones as follows:

$$H^+ \rightarrow W^+ H^T \tag{17}$$
$$(I - J_W^+ J)(I - H^+ H) \rightarrow (I - W^+ W). \tag{18}$$

The condition stated by (14) yields the weight matrix which rotates the range space of J^+ as shown in Figure 2. In that figure, one should observe that the range space of the resulting weighted pseudoinverse of J is geometrically orthogonal to that of the pseudoinverse of H, so that the direction of projection by $(I - J_W^+ J)$ is aligned with the direction of projection by H^+.

Following subsections will explain a variety of algorithms according to the form of rank deficiency of augmented Jacobian. Ultimately, Augmented Jacobian Method will be suggested as the realistic alternative to the algorithmic singularity.

3.1 J_{AUG} has full rank

Now, we assume that the augmented Jacobian J_{AUG} is full rank, which corresponds to the case of no singularities. Then the null space of weight matrix is expressed by

$$\mathcal{N}(W) = \mathcal{N}(J_{AUG}) = \mathcal{N}(J) \cap \mathcal{N}(H), \tag{19}$$

and the new scheme of (16) can be simplified using Lemma 3.1 as following GPM form

$$\dot{q} = J_{AUG}^+ \begin{bmatrix} \dot{p} \\ \dot{h} \end{bmatrix} + (I - J_{AUG}^+ J_{AUG}) z. \tag{20}$$

Since both primary and secondary tasks are simultaneously achieved without errors, the priority between the primary and secondary task is neglected as shown above equation (20).

3.2 Augmented Jacobian Method(AJM)

Strictly speaking, the rank deficiency of augmented Jacobian means either kinematic or algorithmic singularity because the kinematic singularities of J, H in itself are included in rank deficiency of augmented Jacobian. To avoid the singularities, as a matter of fact, it is inefficient to investigate the place where J_{AUG} loses rank. Since the characteristics of Jacobian is dependent on the joint position, the singularity problem is highly nonlinear and is difficult to solve. Moreover, the singularity problem is more serious near singularity. A good alternative to the algorithmic singularity is to make the weight strictly positive definite matrix. Although it more or less contaminates the characteristics of the inverse weighted orthogonality with small positive number ϵ

$$H(J^T J + H^T H + \epsilon I)^{-1} J^T \approx 0, \tag{21}$$

it eliminates an algorithmic singularity in new algorithm of (16). Moreover, it does not affect the execution performance of primary task, though it somewhat degrades the performance of secondary task. Now, we modify the algorithm of (16) as following form

$$\dot{q} = J_W^+ \dot{p} + \left\{ I - J_W^+ J \right\} W^{-1} H^T \dot{h} + \left\{ I - J^+ J \right\} \left\{ I - H^+ H \right\} z, \tag{22}$$

where

$$W = J^T J + H^T H + \epsilon I > 0 \tag{23}$$
$$J_W^+ = W^{-1} J^T (J W^{-1} J^T)^{-1}. \tag{24}$$

Making the weight matrix to be strictly positive definite implies that Augmented Jacobian Method (AJM) of (22) does not include the algorithmic singularity. Also, AJM does not affect the performance of primary task as shown in following equation

$$\dot{p} - J\dot{q} = (I - J J_W^+)\dot{p} - J(I - J_W^+ J)W^{-1} H^T \dot{h}, \tag{25}$$

where $J J_W^+ = I$. In Nakamura's scheme of (6), the damped least square inverse of $\widetilde{H}$ has been utilized to overcome the algorithmic singularity. However, it degrades the performance of primary task. Overcoming

Figure 3: Desired Tasks and Algorithmic Singularity

the algorithmic singularity without affecting the performance of primary task is an important advantage of AJM. Also, AJM is superior to Chiaverini's algorithm in that it brings the smaller error for secondary task.

4   Example

This example solves inverse kinematics of position prior to orientation for 3–R planar manipulators as shown in Figure 3. The primary task tracks desired lines and the secondary task keeps the orientation of initial configuration. In this case, the Jacobian matrix of primary task is given by

$$J = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \end{bmatrix}.$$

$$(26)$$

The vector spanning the null space of Jacobian matrix can be found by the cross product of the first and second rows of (26) as follows:

$$Z = \begin{bmatrix} l_2 l_3 s_3, & -l_3(l_1 s_{23} + l_2 s_3), & l_1(l_2 s_2 + l_3 s_{23}) \end{bmatrix}^T$$

$$(27)$$

where $JZ = 0$. Since the Jacobian matrix of secondary task keeping the orientation of initial configuration is expressed by $H = [\,1,\ 1,\ 1\,]$, the algorithmic singularity occurs as shown in Figure 3 when

$$HZ = 0 \rightarrow l_1 l_2 s_2 = 0 \rightarrow q_2 = 0° \text{ or } 180°. \qquad (28)$$

The condition of $\mathcal{R}(H^T) \bigcap \mathcal{R}(J^T) \neq \emptyset$ is equivalent to that of $HZ = 0$. Total execution time is 2 second and the algorithmic singularity occurs when the time is 0.674 second. If we use Nakamura's algorithm of (6), then the desired tasks lead to the algorithmic singularity at an instant as shown in Figure 3. However, the AJM of (22) and Chiaverini's algorithm of (10) does not cause algorithmic singularity and the performance of primary task is not affected by the algorithmic singularity as shown in Figure 4. Also, Figure 4 show that AJM causes the smaller error than Chiaverini's scheme.
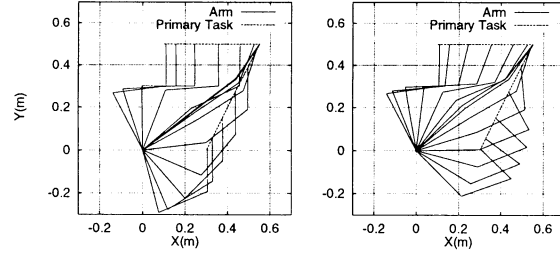
(a)AJM   (b)Chiaverini

Figure 4: Configuration of 3–R Manipulators

5   Concluding Remarks

Through the comparative study for task priority algorithms, we suggested augmented Jacobian approach to task priority redundancy resolution. The core of new algorithm is that we utilize the weighted pseudoinverse in place of pseudoinverse. Then the weight matrix satisfying the inverse weighted orthogonality consists of the augmented Jacobian matrices.

As a realistic alternative to overcome algorithmic singularity, Augmented Jacobian Method(AJM) was suggested in the framework of augmented Jacobian approach. The AJM is to make the weight strictly positive definite matrix without affecting the performance of the primary task. Also, AJM brings the smaller error for the secondary task comparing to previous methods. The efficiency and availability of AJM was shown through simulation.

References

[1] S. Chiaverini, "Singularity–Robust Task–Priority Redundancy Resolution for Real–Time Kinematic Control of Robot Manipulators," *IEEE Trans. of Robotics and Automation*, vol. 13, No. 3, pp. 398–410, 1997.

[2] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics*, John Wiley & Sons, 1988.

[3] Y. Nakamura, H. Hanafusa and T. Yoshikawa, "Task–Priority Based Redundancy Control of Robot Manipulators," *Int. J. of Robotics Research*, vol. 6, No. 2, pp. 3–15, 1987.

[4] J. Park, W. K. Chung and Y. Youm, "Characteristics of Optimal Solutions in Kinematic Resolutions of Redundancy," *IEEE Trans. of Robotics and Automation*, pp. 471–478, 1996.

[5] C. W. Wampler, "Manipulator Inverse Kinematic Solutions Based on Vector Formulations and Damped Least–Squares Methods," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC–16, No. 1, pp. 93–101, 1986.

Mobile Robot Control by Sophisticated Evolution of Multiple CAM-Brains*

Geum-Beom Song and Sung-Bae Cho
Department of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
E-mail:goldtiger@candy.yonsei.ac.kr, sbcho@csai.yonsei.ac.kr

Abstract

There is extensive work to construct a mobile robot controller by evolutionary methods such as genetic algorithm, genetic programming, and so on. However, evolutionary approaches have a difficulty to obtain the controller of complex and general behaviors. In this paper, we propose an incremental evolution method for neural networks based on cellular automata (CA) and a method of combining several modules, evolved as neural networks based on cellular automata, by a rule-based approach. Experimental results show the potential of the incremental evolution and multi-module combination as the techniques for making complex and general behaviors.

1 Introduction

There are some difficulties to construct a sensory-motor controller of mobile robot, such as coordinating the mechanics and control system and managing interaction with an external environment [1]. So, there are many studies of constructing a mobile robot controller by evolutionary approaches such as evolving neural network by genetic algorithm, using genetic programming, and combining fuzzy controller with genetic algorithm. In our previous work [2], we applied the CAM-Brain [3], evolved neural networks based on cellular automata, to controlling a mobile robot. However, this evolutionary method has the difficulty to obtain the controller for complex and general behaviors [4].

In this paper, we propose two methods for sensory-motor controller to do complex behaviors with neural networks based on cellular automata. One is incrementally evolving the controller by starting with simpler environment needed simple behavior and gradually making it more complex and general for those behaviors [4]. The other is combining several modules evolved or programmed to do simple behaviors [5]. The evolved neural networks are based on CAM-Brain, and programmed

module controls the robot directly. We apply a rule-based method to combine the modules and control a mobile robot.

2 Backgrounds

2.1 Behavior-based Robot

Khepera robot (see Figure 1) contains 8 infrared sensors to detect by reflection the proximity of objects in front of it, behind it, and to the right and the left sides of it, and to measure the level of ambient light all around the robot. Also, the robot has two motors to control left and right wheels. Khepera simulator also features the ability to drive a real Khepera robot, so that we can very easily transfer our simulation results to the real robot [2].
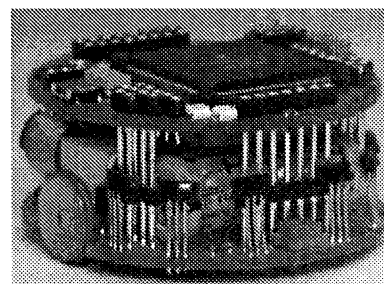
Figure 1: Khepera.

2.2 CAM-Brain

CAM-Brain is neural networks based on CA which can show complicated behavior by combining simple rules. Their structure is decided by chromosome. Therefore, it is possible to evolve neural networks for a specific task by genetic algorithm. It is the set of cells inside 2-D or 3-D CA-space and each cell has one state among blank, neuron, axon and dendrite.

If cell state is blank, it represents empty space and cannot transmit any signals. Neuron cell collects signals

from surrounding dendrite cells, and they are accumulated. If the sum of collected signals is greater than threshold, neuron cells send them to surrounding axon cells. Axon cell sends signals received from neurons to the neighborhood cells. Dendrite cell collects signals from neighborhood cells and passes them to the connected neuron in the end [2, 3].

3 Incremental Evolution

An evolutionary method suffers from the inefficiency to solve complex problem because it is difficult to find proper solution in vast search space. Therefore, it is more feasible to solve a complex problem by evolving the controller with simpler environments and gradually developing it with more general and complex environments. Evaluation tasks $t_1, t_2, t_3, ...,$ and t_n are derived by transforming a goal task in incremental evolution, where n is the number of tasks and t_n is the goal task. In this set, t_i is easier than t_{i+1} for all i: $0 < i \leq n$. Thus, population is evaluated in task t_i and then task t_{i+1} and it finally does in the goal task, t_n [4]. This approach is expected to produce a controller doing complex and general behaviors.

3.1 Method

In this paper, we attempt to incrementally evolve a mobile robot controller to avoid bumping obstacles. Because the robot behavior is composed of going straight and turning left and right, the controller can be incrementally evolved to do those behaviors step by step. The environments in Figure 2(a) and (b) lead the robot to go straight and those in Figure 2(c), (d) and (e) do to turn right, and those in Figure 2(f) and (g) do to turn left.

After deciding environments, we evaluate population of the neural networks in each environment. However, it is important to generate new population which will be used in the next step after completing evolution process in one step. Gomez and Miikulainen [4] solved it by applying delta-coding [6], to escape local minima. In this paper, we select several successful individuals according to fitness and then generate new population from them by applying a modified delta-coding method. The number of mutation in each individual is decided by Cauchy distribution to distribute new individuals properly. Cauchy distribution is defined as follows.

$$f(x) = \frac{\alpha}{\pi(\alpha^2 + x^2)} \qquad (1)$$

With this distribution, the value of f will fall within the interval with 50 percent probability. This way

(a)   (b)   (c)   (d)

(e)   (f)   (g)
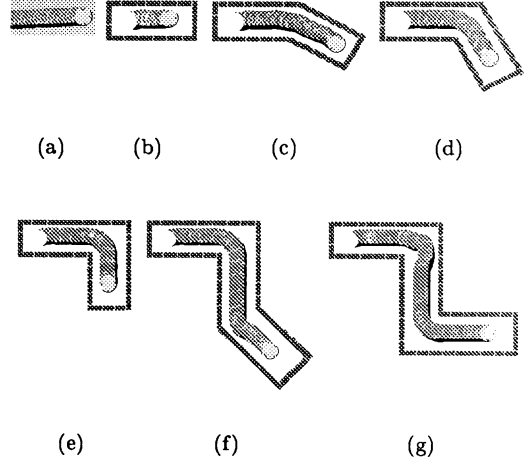
Figure 2: Environments used in incremental evolution.

makes many individuals similar to the individual obtained in the previous step and a few individuals very different to them, because of the shape of Cauchy probability density function. In experiments, we use this distribution to decide the number of mutation on the current population.

3.2 Experimental Results

We use $5 \times 5 \times 5$ CA-space to solve the problem. Fitness is evaluated based on the distance between the present location of the robot and the goal point, and the number of straight movement.

$$Fitness = \frac{50}{D} \times (\frac{1}{S} \sum_{i=0}^{S} V_i) \qquad (2)$$

D : Distance between the robot and goal position
S : The number of movement until stopping
V_i : The value decided by the velocity in the ith step

This fitness evaluation leads the robot to arrive at goal position quickly without bumping against walls. Population size is 50 and the fitness of individual is computed by the average of four runs. Figure 3 shows the change of the best fitness when evolving in the environment of Figure 2(g). Figure 4 shows it when using the incremental evolution and Cauchy distribution. In this figure, we can see the efficiency of incremental evolution for complex problem by comparing the existence of the successful individual. Figure 5 shows the architecture of the evolved neural networks in the final step. We can see that the architectures of neural networks get complicated as the step goes on. Figure 6 shows the results of applying the individual obtained by the incremental evolution method to more complex environments.
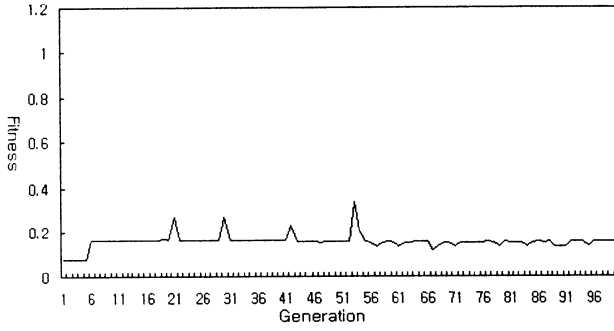
Figure 3: The change of the best fitness when evolving in Figure 2(g) environment.
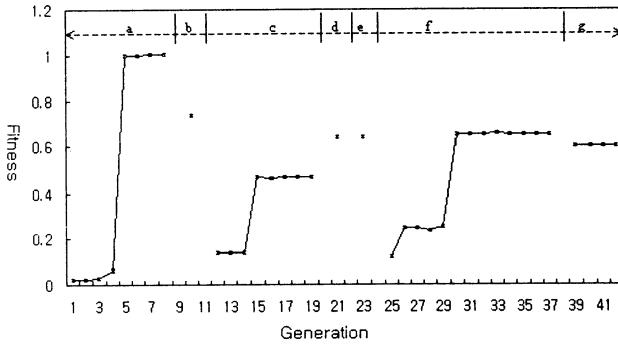
Figure 4: The change of the best fitness with the incremental evolution with automatic selection of the best individuals.

4 Multi-modules Integration

The controller composed of one module has a difficulty to make the robot to perform complex behavior. To overcome this shortcoming, some researchers combine several modules evolved or programmed to a simple behavior such as "going straight," "avoiding obstacles," "seeking object," and so on. They expect the controller combined with several modules can do complex behaviors.

4.1 Method

Combinig low and simple behaviors make high-level complex behaviors. In this section, basis behaviors and the IF-THEN rules for combining them are presented. Four basis behaviors used in this paper are defined as follows.

- Battery Recharge : If a robot arrives at battery recharge area, battery is recharged. This module enables the robot to operate as long as possible.

- Following Light : The robot goes to stronger light. It must operate this module to go to the "Battery
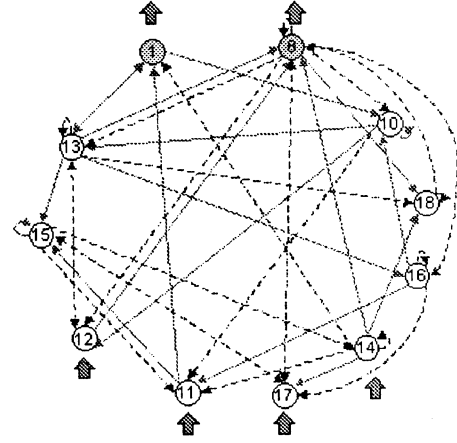
Figure 5: The neural network architectures in the final step (doted line: inhibitory connection, solid line: excitatory connection).

(a)   (b)
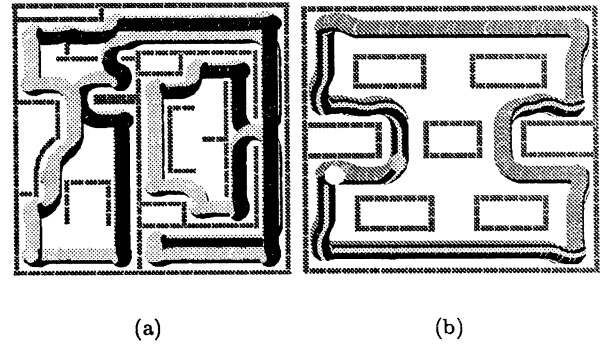
Figure 6: Applying the successful robot to different environments.

Recharge" area because the light source exists in that area.

- Avoiding Obstacles : If the obstacles exist around the robot, it avoids obstacles without bumping against them. This module enables it to go to "Battery Recharge" area safely by avoiding obstacles.

- Going Ahead : If there is nothing around the robot, it goes ahead. This module makes it to move continuously without stopping.

These basis behaviors are needed to adapt to given environments shown in Figure 7. In order to live for a long time in a given environment the robot avoids bumping against obstacles and goes to the "Battery Recharge" area for recharging battery occasionally.

We use IF-THEN rules for combining the four basis behaviors properly. Using IF-THEN rules, we can decide an operating module according to the situation
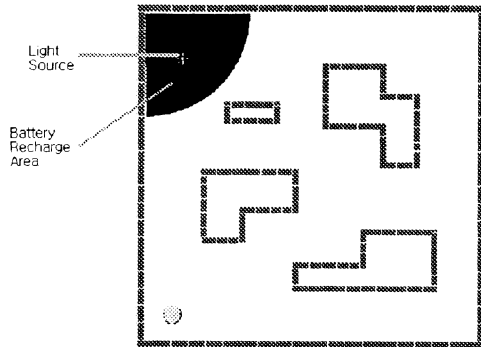
Figure 7: Simulation environment.

(a)   (b)

Figure 8: The trajectories of the robot with multi-modules.

which is judged by sensor values of the robot. We expect that this approach adapts to a given environment if the rules are defined properly. The rules used in this paper are as follows.

IF ("Battery Recharge" area)
    Execute Battery Recharge module
ELSE
    IF (Battery sensor $< \alpha$) AND
        (Minimum value of light sensors $\leq \gamma$)
        IF (Maximum value of distance sensors $\leq \beta_1$)
            Execute Following Light module
        ELSE
            Execute Avoiding Obstacles module
    ELSE
        IF (Maximum value of distance sensors $\leq \beta_2$)
            Execute Going Ahead module
        ELSE
            Execute Avoiding Obstacles module

We can see the process of operating modules because these rules are very simple and clear. The robot moves differently according to each constant values (α, β, γ).

4.2 Experimental Results

We simulate this approach in a modified Khepera simulator. We select the operating module by the rules explained at the previous section and sensor values of the robot. Figure 8 shows the trajectory of the robot by combined controller in a given environment. According to starting position of the robot, the trajectory and moving behavior are different.

5 Concluding Remarks

This paper has attempted to solve the difficulty to obtain the controller doing complex and general behavior by incremental evolution and multi-module inte-

gration. Experimental results of incremental evolution method show the potential making complex and general behaviors efficiently. Also, we has combined multi-modules evolved or programmed to do simple behaviors using IF-THEN rules to generate complex behaviors, so we show the feasibility of combining multiple CAM-Brain modules.

References

[1] S. Nolfi, D. Floreano, O. Miglino and F. Mondada, "How to evolve autonomous robots: Different approaches in evolutionary robotics," *Proc. of the Int. Conf. on Artificial Life IV*, pp. 190-197, 1994.

[2] G.-B. Song and S.-B. Cho, "Applying evolved neural networks based on cellular automata to robot control," *Proc. of 5th Int. Conf. on Soft Computing*, pp. 837-840, Japan, 1998.

[3] F. Gers, H. de Garis and M. Korkin, "CoDi-1Bit: A simplified cellular automata based neuron model," *Proc. of Artificial Evolution Conf.*, France, 1997.

[4] F. Gomez and R. Miikkulainen, "Incremental evolution of complex general behavior," *Adaptive Behavior*, Vol. 5, Issue 3-4, pp 317-342, 1997.

[5] W. Banzhaf, P. Nordin and M. Olmer, "Generating adaptive behavior using function regression within genetic programming and a real robot," *Int. Conf. on Genetic Programming*, pp. 35-43, 1997.

[6] K.E. Mathias and L.D. Whitley, "Initial performance comparisons for the delta coding algorithm," *Proc. of IEEE Conf. on Evolutionary Computation*, Vol. 1, pp. 433-438, 1994.

Progressive Evolution with Subgoals that are Common Properties in Elite Population

Motoaki Matsuzaki, Takamitsu Kawai, Hideki Ando, and Toshio Shimada
Department of Information Electronics, Graduate School of Engineering
Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan

Abstract

A new progressive evolution method getting subgoals autonomously is proposed to efficiently generate logic circuits whose specifications are unknown. This progressive evolution method defines the next subgoal as the common properties in the current elite population. We applied this method to the action control circuit for artificial ants as an example of unknown-specification logic circuits to confirm effectiveness. Our progressive evolution method extends intrinsic searching ability in conventional GAs utilizing subgoals effectively. Our experimental results show that better convergency can be achieved than conventional GAs.

1   Introduction

Evolvable hardware[1][3] can change its architecture and behavior autonomously according to given environment. The two key technologies to realize evolvable hardware are **genetic algorithms (GAs)**[6] and reconfigurable logic devices such as field programmable gate arrays (FPGAs). Evolvable hardware acquires necessary structures or behavior by coding the logic circuit as genes and learning with GAs. Evolvable hardware is expected to be applied to various complex applications that are difficult to give its logical specification or optimal architecture, such as control circuits for anti-objects robots and branch predictors of microprocessors.

Setting subgoals during evolution is useful in designing logic circuits whose specifications are unknown to build the necessary structures efficiently. Hikage introduced a **progressive evolution model**[5] that sets the subgoal for the structure to be learned and acquire necessary structures progressively. In the progressive evolution model, the subgoals are given by the designers. Although this method speeds up the evolution by dividing a large search space into some small

spaces, the designers must set the subgoals with great efforts for each problems. Therefore it is not practical for very complex problems.

In this paper, we introduce a progressive evolution method for automatic logic circuit generation, in which the evolution gets subgoals autonomously.

2   Related Work

Hikage introduced a progressive evolution model[5] for efficient evolution in designing logic circuits whose specifications are unknown, and applied it to generate the action control circuit of artificial ants. The progressive evolution model changes its objectives from easy ones to hard ones. This model divides a problem "hurdle" into small steps. This model requires some subgoals to get necessary structures stepwise in logic circuit generation. The learning method of the model is as follows: repeat a procedure that gives a subgoal and learns for the subgoal until learning for the final goal finishes. For example, in generating action control circuits for artificial ants, initially we give an environment described as a map (subgoal) for learning to acquire the circuit structures that makes outputs as follows; walk forward if a food exists in front of the ant or no food exists around it, and turn right if no food exists in front of it and a food exists on its right-hand side. When learning for the previous subgoal is done, the next environment to learn additional action patterns is given. Finally we give the target map as the finel environment.

However, because the progressive evolution model by Hikage is not autonomous, designers must make great efforts to set the subgoals to each problem.

To avoid such efforts, we introduced **autonomous progressive evolution**[11] that is the progressive evolution that autonomously gets subgoals, and applied it to generate the action control circuit for artificial ants. In this method, an individual that has high-

est fitness value in the current generation becomes the current subgoal. This method performs both global search for the final goal and local search for the current subgoal concurrently. In the local search for the current subgoal, the population learns the action patterns of the current subgoal. Since the local search for the current subgoal itself may lose diversity of the population, we define the "neighbourhood" of the current subgoal, and make the local search move individuals toward the "neighbourhood." The area of the "neighbourhood" is determined by the degree of achievement in the search. The area of the "neighbourhood" becomes smaller as the subgoal approaches to the final goal.

In autonomous progressive evolution, however, there are no general ways to obtain the "neighbourhood" function. It depends on experience or intuition of system designers, like setting method of the genetic parameters or the fitness functions in general GAs.

3 Progressive evolution with subgoals that are common properties in an elite population

The GAs that have fixed-bit-length genes such as simple GA can be easily applied to progressive evolution by building block. However logic circuit generation with GAs often have more complex variable gene structures such as tree structures rather than fixed-length bit strings. In general, it is difficult to form the building block with these complex structures. We propose a new progressive evolution method getting subgoal autonomously for any kind of GAs that have complex gene structures.

In this section, we show the outline of the proposal method and additional fitness functions for the method.

3.1 Outline

The learning target of general GAs is just the final goal. It takes very long time to find one special goal point in a large space by using only global search. On the other hand, progressive evolution leads the population to the final goal stepwise with some suitable subgoals. The advantage of progressive evolution is that population can reach the final goal in short time by step-by-step local searches in the divided search space.

The very important factor for successful evolution is how to set the subgoals. Also the method for design-

ing the fitness function for local search with subgoals should be general. We consider the subgoal as follows:

- In the neighbourhood of the subgoal, there may be some points closer to the final goal than the subgoal.

- The extent of the neighbourhood of the subgoal becomes smaller as the subgoal becomes closer to the final goal.

On the other hand, the population has the following characteristics:

- The elite population (individuals with higher fitness values) of current generation has common useful properties.

Considering above discussion, we define the next subgoal as the common properties in the current elite population, and to evaluate individuals in local search, we map each individual onto the associated space where the properties of the individuals are comparable one another (we call this associated space "**property space**"). The images in the property space are fixed-length data strings like a gene in simple GAs, and it is enumeration of the actions of an individual for its inputs (we call this "**property**"). In this method, a schema is formed from the images of an elite population in the property space in every generation. This schema is the subgoal in the current generation. And we call these schemata in the property space "**sub-schemata**." The later the generation of the evolution is, the larger the order of the sub-schema is. In the earlier generation, the population performs search in the lager space in local search. As generations go by, the subgoal approaches the final solution and the searching space is reduced, thus local search can be performed successfully.

We propose a new autonomous progressive evolution method as follows:

1. The subgoal of the next generation is the common properties of the elite population in the current generation.

2. Both global search for the final goal and local search for the current subgoal are concurrently performed.

The population acquires the subgoals autonomously by the first item. And the second item means that the population searches both the final goal and the subgoal. Thus the evolution performs both global search for the final goal and local search for the subgoal concurrently. Note that each individual has two fitness

values; one is for the final goal, the other is for the subgoal. These two fitness functions have some relationship with each other. Additionally the second item makes the subgoal be close to the final goal. Concretely, the population consists of individuals made by the genetic operations using the fitness values for the final goal and individuals made by the genetic operations using the fitness values for the subgoal.

3.2 The regular fitness and the subgoal fitness

As the previous discussion, we define two kinds of fitness values to implement the autonomous progressive evolution. One is a **regular fitness value**, the other is a **subgoal fitness value**.

The regular fitness value $F_p(i)$ is the value indicating how close to the final goal the individual i is. This is equal to the fitness value of general GAs. The subgoal fitness value $F_s(i)$ is the fitness value of the individual i for the subgoal. $F_s(i)$ is computed using the image $P(i)$ in the property space. The mapping function P is the function which maps the individual onto the property space, or from the gene structures onto data strings of enumeration of behavior or characteristics of the individuals. Before the computation of the subgoal fitness value, we obtain the subgoal of the current generation as follows: compute regular fitness values of all the individuals and sort them, then abstract the common properties $P(i)$ of N_{elite} individuals from the top, thus we can make a sub-schema. The way to make sub-schemata is just the same as simple GAs. The data strings of sub-schemata consist of the combination of values of properties and the don't-care characters "*." Let the sub-schema made here be the current subgoal P_{sg}.

The subgoal fitness function F_s that gives subgoal fitness value $F_s(i)$ is the function of the image $P(i)$ and the subgoal P_{sg}:

$$F_s(i) = f(P(i), P_{sg}) \qquad (1)$$

3.3 Progressive evolution with subgoals that are common properties in an elite population

The procedure of the progressive evolution with subgoals that are common properties in an elite population is as follows:

1. Make an initial population randomly

2. Compute regular fitness values $F_p(i)$ of each individual i

3. Sort the population with respect to $F_p(i)$ in descending order, then make the current subgoal P_{sg} based on the properties $P(i)$ of N_{elite} individuals from the top.

4. Make properties $P(i)$ of each individual i, and compute the subgoal fitness values $F_s(i)$

5. Select parents by the selection operation with $F_p(i)$

6. Select parents by the selection operation with $F_s(i)$

7. Make offsprings by genetic operations from the parents group made by step 5 and 6

8. Go back to step 2

Note that selected individuals by step 5 and 6 are stored into the same group. Namely there are no sub-populations.

4 Experiments of action control circuits for artificial ants

4.1 Artificial ant problem

A map for artificial ant problem is shown in Figure 1. Each gray cell contains a food and the ant traces the foods. There is no food on the white cells. If the ant traces on a food, the food is eaten and disappears. The map has a torus structure where its boundaries are connected on the left and right side, and the upper and lower side, respectively. The solid line depicts a valid path for the ant to get their foods.

Our goal is to make an action control circuit that control the ant so that it can get all the foods on the map shown in Figure 1. This circuit has a 5-bit input and a 2-bit output. The ant can see whether there is a food or not on the cell of the forward, left, right, left-forward, and right-forward side shown in Figure 2. The 5-bit input corresponds to the existence of the food. The 2-bit output corresponds to 3 actions: go forward, turn left, and turn right.

The fixed number of foods are placed on the map based on a certain rule. Therefore in this problem the action rule is obtained in a form of logic circuits, according to the rule of placement of the foods. For each of 32 input patterns, one of the 3 actions is assigned. Therefore there exist 3^{32} combinations thus searching all pattern is very difficult.
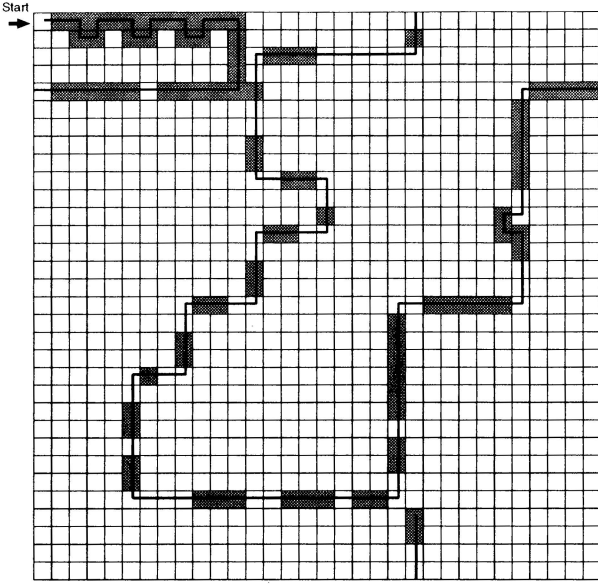
Figure 1: Artificial ant problem for experiment

Table 1: Relationship between outputs and actions of the artificial ant

Out0	Out1	Action
0	0	Turn Left
1	0	Turn Right
0	1	Walk Forward
1	1	Walk Forward

4.2 Experiment of logic circuit generation

4.2.1 Tree-structured representation of logic circuit

We assume the representation of the logic circuit to be a tree structure shown in Figure 3.

In Figure 3, the gene representation of the logic circuit (a) is (b). One individual is represented in a tree and the tree is constructed based on a kind of generative grammar.

Starting from the start symbol **ROOT**, the output nodes **OUT1s** are generated as many as the number of the outputs. The nodes below the output nodes stand for logical symbols (e.g. **AND2s**, **NOT1s**), and the terminal nodes represent the external input (**EXTINs**). The arcs between each nodes represent their wiring.
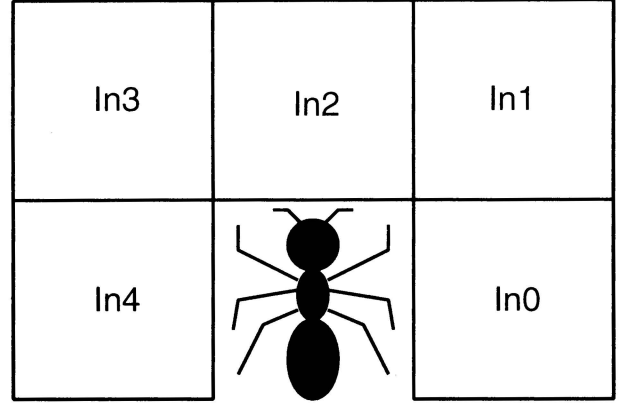
Figure 2: Inputs for artificial ant

4.2.2 Evaluation function

We get the regular fitness value by letting the ant including corresponding individual walk through the map actually. We regard one action (go forward, turn left (or right) at the place) of the ant as 1 step, and the number of steps is limited up to 350. We let the regular fitness function be

$$F_p(i) = \beta * \#foods(i) + \gamma * \#nodes(i), \qquad (2)$$

where $\#foods(i)$ is the number of obtained foods, and $\#nodes(i)$ is the number of nodes in the individual. β and γ are weighting coefficients and we set them as $\beta = 100$, $\gamma = 0.01$.

The reason why we add the number of the individual in the regular fitness function F_p is because we prevent the individual with high $F_p(i)$ that have the significantly smaller number of nodes than the number of nodes in desired circuit configuration from falling in the local minimum.

We let the mapping function P onto property space to get subgoal fitness function be input/output relationship of individuals. We let the subgoal fitness function, using the hamming distance d_{hum} between subgoal P_{sg} and $P(i)$ be

$$F_s(i) = 32 - d_{hum}(P(i), P_{sg}), \qquad (3)$$

where "32" is the total number of the input/output relationship of generated circuits, provided that we deal with "*" element(s) in subgoal P_{sg} as "not-matched."

4.2.3 Experiment of logic circuit generation

We carried out experiments to confirm the effectiveness of subgoal generation. The measured success ratio
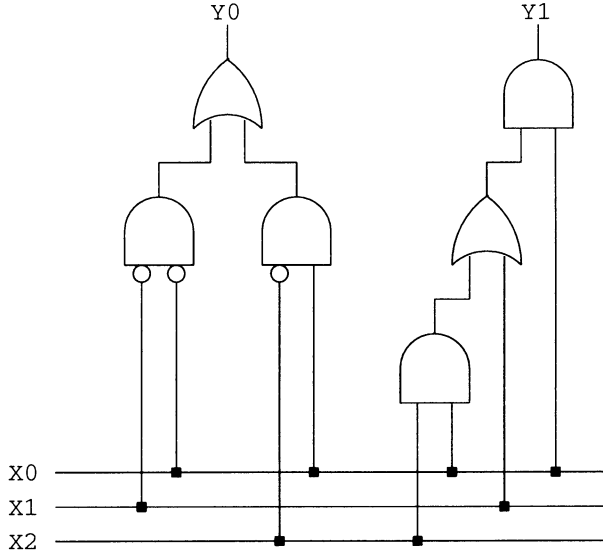
(a) Logic circuit
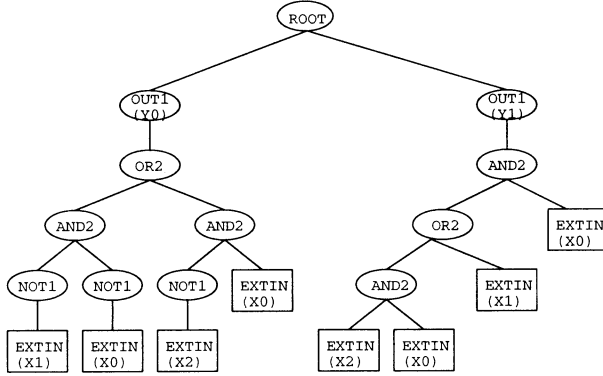
(b) Tree-structured representation

Figure 3: An example of gene representation

Figure 4: Success ratio of control circuit for an artificial ant
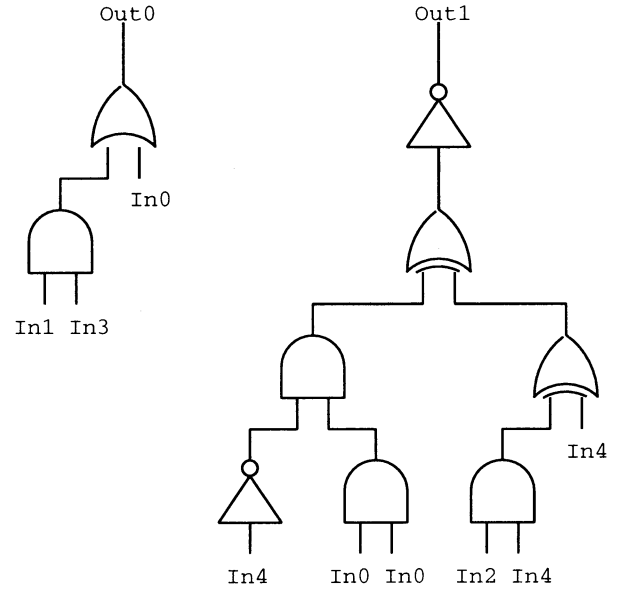
Figure 5: Generated control circuit for an artificial ant

is shown in Figure 4. The dotted line shows the case without subgoal generation(N/A), and the solid line shows the case with subgoal generation(w/subgoal). We used the same initial gene group.

We used the elite preservation strategy in each experiment. In each generation, 1 elite individual is added to the population. In each experiment, we used 100 individuals, 60% as crossover probability, and 1% as mutation probability. We applied these experiments to 100 kinds of initial gene groups.

We can see that w/subgoal is totally better.

The circuit shown in Figure 5 is an action control circuit of an ant by a certain trial using w/subgoal. Note that this circuit is valid though it has some redundancy parts such as AND gate with the same in-

5   Conclusion

Automatic logic circuit generation is one of indispensable technologies to realize evolvable hardware. We proposed a new progressive evolution method getting subgoals autonomously to efficiently generate logic circuits whose specifications are unknown. This

progressive evolution method defines the next subgoal as the common properties in the current elite population. We applied this method to the action control circuit for artificial ants as an example of unknown-specification logic circuits. Our progressive evolution method extends intrinsic searching ability in conventional GAs utilizing subgoals effectively. We confirmed that our method achieved better convergency than conventional GAs.

Acknowledgements

References

[1] H. Hemmi, J. Mizoguchi, and K. Shimohara, "Development and Evolution of Hardware Behaviors," R. Brooks, P. Maes, *Artificial life IV*, pp. 371-376, MIT Press, 1994.

[2] H. Hemmi, J. Mizoguchi, and K. Shimohara, "Development and Evolution of Hardware Behaviors," *Towards Evolvable Hardware (Lecture Note in Computer Science 1062)*, pp. 250-265, Springer, 1996.

[3] T. Higuchi, T. Niwa, T. Tanaka, H. Iba, H. de Garis, and T. Furuya, "Evolvable Hardware with Genetic Learning," In *Proc. of Simulated Adaptive Behavior (SAB92)*, MIT Press, 1992.

[4] T. Higuchi, M. Iwata, I. Kajitani, H. Iba, Y. Hirano, T. Furuya, and B. Manderick, "Evolvable Hardware and its Applications to Pattern Recognition and Fault-tolerant Systems," *The First International Workshop "Towards Evolvable Hardware,"* 1995.

[5] T. Hikage, H. Hemmi, and K. Shimohara, "Progressive Evolution Model Using A Hardware Evolution System," In *Proc. of International Symposium on Artificial Life and Robotics (AROB)*, 1997.

[6] J. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan, 1975, and MIT Press, 1992.

[7] W. Liu, M. Murakawa, and T. Higuchi, "ATM Cell Scheduling by Function Level Evolvable Hardware," In *Proc. of the First International Conference on Evolvable Systems*, Springer-Verlag, 1997.

[8] M. Murakawa, S. Yoshizawa, and T. Higuchi, "Adaptive Equalization of Digital Communication Channels Using Evolvable Hardware," In *Proc. of the First International Conference on Evolvable Systems*, Springer-Verlag, 1997.

[9] J. R. Koza, *Genetic Programming*, MIT Press, 1992.

[10] J. R. Koza, *Genetic Programming II*, MIT Press, 1994.

[11] M. Matsuzaki, T. Kawai, H. Ando, and T. Shimada, "Automatic Logic Circuit Design with Progressive Evolution that Autonomously Gets Subgoals(in Japanese)," *Trans. of the Society of Instrument and Control Engineers*, Vol. 35, No. 4, pp. 560-567, 1999.

A Proposal of Hardware GA Engine Compiler using Extended C Language and Field Programmable Gate Array

Makoto FUJII   Takeshi FURUHASHI

Dept. of Information Electronics, Graduate School of Eng., Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-01, JAPAN
E-mail:{fujii, furu}@bioele.nuee.nagoya-u.ac.jp

Abstract

Genetic Algorithm (GA) is one of good tools to solve combinatorial optimization problems. This method is expected to solve many complex problems efficiently. However, much computation time is still needed for GA search. GA operations consist of many bit operations. Hardware implementation of GA operations brings about more speedups. GA operations depend on each application, so stereotyped hardware implementation of GA operations is difficult.

This paper proposes a hardware GA engine compiler using extended C language. The extended C language can describe bit operations directly. This compiler analyzes a program written by the extended C language and translates a program into VHDL (Hardware Description Language), and then, implements the program in hardware using FPGA (Field Programmable Gate Array). Two simulations, ALL-1 problem and scheduling problem, are done to show the efficiency of this compiler. As a result, the speed of hardware GA engine (ALTERA MAX PLUSII, 20MHz) generated by this compiler is 42~43 times faster than a workstation (Sun Ultra-SparcII 300MHz), and 15~17 times than PC (Pentium III 450MHz).

1   Introduction

Genetic Algorithm (GA) is one of the good tools to solve combinatorial optimization problems. Based on evolution of life, it can solve many complex problems efficiently. However, much computation time is still needed to execute GA search using software on a general-purpose processor (CPU). There have been researches on hardware GA engine. P. Graham and B. Nelson proposed "SPGA"(Splash2 Parallel Genetic Algorithm) to solve TSP problems on Splash2[1][2]. The SPGA had 4-FPGA's and 4-memory chips and the maximum clock frequency was 11MHz. The SPGA was 2~4 times faster than a workstation (HP PA-RISC 125MHz). N. Sitkoff et al. proposed "Armstrong III" system based on PRISM II architecture[3]. Some components of GA were implemented across two or more FPGAs while minimizing the number of external connections. "Armstrong III" was 3 times faster than a workstation (SUN SPARCstation20 60 MHz). S. D. Scott et al. pro-

posed hardware-based genetic algorithm "HGA"[6]. HGA took only 6% as many clock cycles to run as a software. These studies have translated popular GA operations into hardware operations, thus speedups against software operations are not significant.

There have been some studies on hardware GA engine using specially designed GA operations for hardware[4]. O.Kitaura et al. proposed "H^3" GA engine without pipeline stalls. H^3 employed the steady-state GA instead of the simple GA which was employed in most previous GA engine. H^3 engine was about 700 times faster than a workstation (DEC Alphastation 333MHz). N.Yoshida and T.Yasuoka proposed VLSI hardware "Multi-GAP" extended "GAP" system, which also employed the steady-state GA[5]. These hardware GA engines ran very faster than software because these engines employed suitable GA operations (the steady-state GA) for hardware implementation. But some operations in these hardware GA engine were limited to specific ones. For example, a selection part employed only the roulette wheel selection.

This paper proposes a hardware GA engine compiler using extended C language. The extended C language can describe bit operations directly. This compiler analyzes a GA program written by the extended C language and translates it into VHDL (Hardware Description Language), and then, implements the program into hardware using FPGA (Field Programmable Gate Array). Two simulations, ALL-1 problem and scheduling problem, are done to show the efficiency of this compiler. As a result of evaluation, the execution time of hardware GA engine(ALTERA MAX PLUSII, 20MHz) generated by this compiler is 42~43 times faster than that of a workstation (Sun Ultra-SparcII 300MHz), and 15~17 times than a PC (Pentium III 450MHz).

This paper is organized as follows. Section 2 defines the extended C language. Section 3 explains the process of compilation of the proposed hardware GA engine compiler. Section 4 evaluates the hardware GA engine generated by the proposed compiler. Finally, section 5 concludes this paper.

```
/*   C code to count the number of '1' of each bit in gene   */

       int  i , gene , number ;

       number = 0 ;

       for ( i = 0 ; i < 16 ; i++ ) {

           number + = gene & 0x1 ;  /*  masking */
           gene = gene >> 1 ;           /*  right 1-bit shift */

       }
```

Figure 1: C code to count the number of '1' of each bit in *gene*

```
/*   Extended C code to count the number of '1' of each bit in gene   */

       int  i , gene , number ;

       number = 0 ;

       for ( i = 0 ; i < 16 ; i++ ) {

           /*  description of using bit operation function */
           number += bit_operation ( gene , i ) ;

       }
```

Figure 2: Extended C code to count the number of '1' of each bit in *gene*

2 Extended C Language

2.1 Why does C language have to be extended?

GA operations consist of repetition of many bit operations, and it is efficient to execute bit operations using digital hardware. C language is one of the most popular software languages. C language can handle bit operations, but can't represent bit operations directly. Figure 1 shows a GA program written by C language, which counts the number of '1' of each bit in a *gene*. To count the number of '1' of each bit in the *gene*, masking and right 1-bit shift are used. In software implementation, these operations are needed.

In hardware implementation, these operations are needless because each bit in the *gene* can be handled independently. If C language can represent bit operations suitable for the hardware implementation without using the masking and right 1-bit shift, efficient hardware implementation of GA operations can be achieved. Next paragraph proposes Extended C Language to represent the bit operations.

2.2 Definition of Extended C Language

The extended C language is based on the ordinary C language. Some functions to represent the bit operations are added.

- Hardware generating function: GA_hardware()
  Hardware generating function is the pointer to the implementation of hardware compilation in a GA program. All GA operations written in this function are translated into hardware.

- Bit operation function: bit_operation(*op1,op2*)
  op1 is a variable that executes bit operations, *op2* is the position of bit operations in *op1* variable. Bit operation function considers a bit as a variable and handles the bit directly.

- Library function : library(*op1*)
  op1 is a library name. This function is used for operations specified or used many times in a program (random value generator etc.), and is translated into hardware using hardware library.

Figure 2 shows a GA program, which counts the number of '1' of each bit in the *gene* written by the extended C language. Using the bit operation function, the program in Figure 1 can be rewritten as Figure 2. Each bit in the *gene* is represented as a variable and the masking and right 1-bit shift operations are removed. As a result, the extended C compiler can translate bit operations into efficient hardware.

3 Compilation

The proposed hardware GA engine compiler translates a GA program written by the extended C language into hardware as follows. Figure 3 shows the procedure of the compilation.

1. Searching
   The compiler scans a GA program written by the extended C language and searches the hardware generating function "GA_hardware()".

2. Making a bit-level control/data flow graph
   Next, the compiler scans and analyzes GA operations in GA_hardware() function, and makes a bit-level control/data flow graph.

3. Generating calculation units of hardware GA engine
   From a bit-level data flow graph, the compiler analyzes the number of operations to be able to execute at the same time and generates calculation units to impliment the hardware. Library functions are translated directly using hardware library.

4. Generating a control unit of hardware GA engine
   From a bit-level control/data flow graph, the compiler generates a control unit of hardware GA engine.
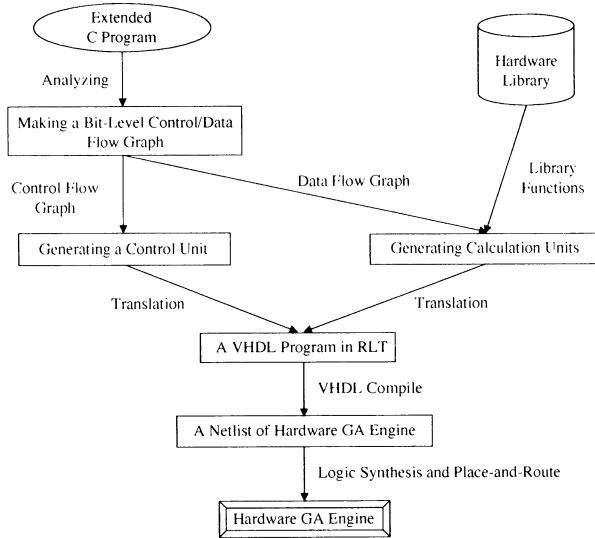
Figure 3: The procedure of hardware GA engine compilation

The hardware GA engine executes GA operations in accordance with this unit's instructions.

5. Translation into VHDL and generation of a netlist
   The compiler generates a VHDL program of hardware GA engine from the above data, and VHDL compiler translates it into a netlist.

6. Logic Synthesis and Place-and-Route
   Finally, FPGA compiler synthesizes and routes hardware logic, the hardware GA engine is generated and compilation is ended.

4 Evaluation of execution time

The proposed hardware GA engine compiler in this paper was applied to two GA programs, which were ALL-1 problem and task scheduling problem. These two GA programs included many bit operations not only crossover/mutation operations but also calculation of a fitness value. We compared execution time of the hardware GA engine generated by the proposed compiler with that of software implementation. To measure execution time of hardware GA engine, we used timing simulation of the ALTERA MAX PLUSII FPGA-CAD and the clock frequency of hardware GA engine was 20MHz. In software implementation, we wrote a GA program in ordinary C language and compiled using gcc. The software was run on the workstation and the PC and the profile was measured with the UNIX gprof command. The workstation was a Sun compatible workstation with 300MHz Ultra-SparcII processor with SunOS V5.6 and the PC was an IBM PC/AT compatible PC with 450MHz Intel Pentium III processor with Linux V2.2.6.
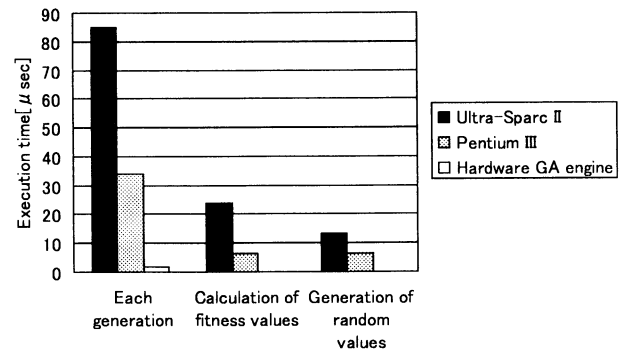
Figure 4: Evaluation of execution time in each generation on ALL-1 problem

4.1 ALL-1 Problem

ALL-1 problem is a very simple problem. When all bits in a gene become '1', a fitness value also becomes the largest in the simulation environment. In this simulation, the number of individuals was 16, the bit length of each gene was 16 bits, and the fitness function was the summation of value of each bit in a gene. GA operations in each generation were: (1) Calculation of a fitness value of each genes, (2) Selection of 8 individuals whose fitness values were high, (3) Crossover and reproduction using selected 8 individuals as parents, (4) Mutation of each bit of all individuals by 6.25%.

Table 1: Evaluation of execution time in one generation with ALL-1 problem[μ sec]

	α	β	γ
Ultra-Sparc II	85.1	23.9	13.1
Pentium III	33.9	6.22	6.33
Hardware GA engine	1.95	0.05	0.05

α : Each generation
β : Calculation of fitness values
γ : Generation of random values

Table 1 and Figure 4 show evaluation of execution time in ALL-1 problem. In each generation, execution time of hardware GA engine generated by hardware GA engine compiler was 43.6 times faster than that of the workstation and 17.4 times than that of the PC. In calculation of the fitness variable, the hardware GA engine was 478 times faster than the workstation and 124 times than the PC. In random value generation, the hardware GA engine was 262 times faster than the workstation and 127 times than the PC. much time was needed in calculation of a fitness value and generation of a random value. By translating these operations into hardware using bit operation function and library function directly, the hardware GA engine became
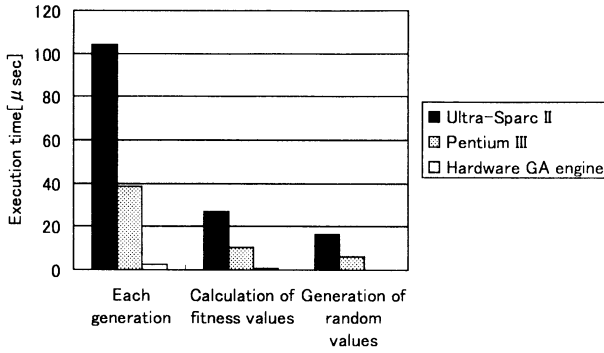
Figure 5: Evaluation of execution time in each generation on task scheduling problem

faster than the software although the clock frequency was only 4.4~6.6% of these computers.

4.2 Task Scheduling Problem

Next, the proposed compiler was applied to a task scheduling problem. Task scheduling problem is a problem often used to measure the performance of GA. In this simulation, 8 tasks were assigned to 4 machines, whose performances were different. GA parameters and operations were the same as those for the ALL-1 problem.

Table 2: Evaluation of execution time in each one generation on task scheduling problem[μ sec]

	α	β	γ
Ultra-Sparc II	104	27	16.3
Pentium III	38.9	10.5	6.02
Hardware GA engine	2.45	0.60	0.05

α : Each generation
β : Calculation of fitness values
γ : Generation of random values

Table 2 and Figure 5 show execution time in solving the task scheduling problem. In each generation, the execution time of the hardware GA engine generated by the hardware GA engine compiler was 42.6 times faster than that of the workstation and 15.6 times than the PC. In calculation of the fitness variable, the hardware GA engine was 45 times faster than the workstation and 10.5 times than the PC. In random value generation, the hardware GA engine was 325 times faster than the workstation and 120 times than the PC. The hardware GA engine was superior in speed to software.

5 Conclusions

This paper presented the hardware GA engine compiler using the extended C language that could represent bit operations suitable for hardware implementation. Two GA programs(ALL-1 problem and task scheduling problem) were used to evaluate the hardware GA engine generated by the proposed compiler. The generated engines were 42~43 times faster than a workstation(Sun UltraSparcII 300MHz), and 15~17 times than a PC(Pentium III 450MHz).

Acknowledgement

References

[1] P.Graham and B.Nelson,"A Hardware Genetic Algorithm for the Traveling Salesman Problem on Splash2", In *Proceedings of the 5th Field Programmable Logic and Applications*, pp.352 361, 1995.

[2] P.Graham and B.Nelson,"Genetic Algorithms In Software and In Hardware A Performance Analysis Of Workstation and Custom Computing Machine Implementations", In *Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines*, pp.216 225, 1996.

[3] N.Sitkoff, M.Wazlowski, A.Smith and H.Silverman, "Implementing a Genetic Algorithm on a Parallel Custom Computing Machine", In *Proceedings of IEEE Workshop on FPGAs for Custom Computing Machine*, pp.180 187, 1995.

[4] O.Kitaura, H.Asada, M.Matsuzaki, T.Kawai, H.Ando and T.Shimada, "A Custom Computing Machine for Genetic Algorithms without Pipeline Stalls", In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics(SMC'99)*, Vol.V, pp.577 584, 1999.

[5] N.Yoshida and T.Yasuoka, "Multi-GAP: Parallel and Distributed Genetic Algorithm in VLSI", In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics(SMC'99)*, Vol.V, pp.571 576, 1999.

[6] S.D.Scott, A.Samal and S.Seth, "HGA: A Hardware-Based Genetic Algorithm", In *Proceedings of the 1995 ACM/SIGDA 3rd International Symposium on FPGA*, pp.53 59, 1995.

A New Genetic Encoding for Evolvable Hardware

— on the Dilemma between Unconstrained approach and Effective search —

Akinori NAKATA   Masaharu MIZUMOTO

Division of Information and Computer Sciences
Osaka Electro-Communication University
Neyagawa, Osaka, 572-8530, Japan

Abstract

In this paper, discussing "Unconstrained" approach to design, especially, on Evolvable Hardware (E-HW), Evolving Medium dependent Genetic Encoding (EMdGE) is suggested as one of improving methods against the dilemma between unconstrained intrinsic evolutionary approach and effective search in design space on EHW. Furthermore, we show how to use this method for autonomous mobile robot.

1   Introduction

Adrian Thompson et al. [1] proposed "Unconstrained Intrinsic Evolvable Hardware". Evolutionary Algorithms (EAs) can explore some of the regions, and produce new designs of electronic circuits better than the conventional design methods can do, when EAs can explore in the design space as freely as possible.

Let us reconsider the meaning of "Unconstrained". Generally speaking, when we create something, we image how it works, how strong it needs, how big it is, and so on. Then, we gather its well-suited stuffs, and produce it. This process intuitively shifts our product from abstract level to concrete level. The more we select something to become it concrete, the more it is constrained to selected ability, structure and so on, even though we can see it becomes concrete more and more. Rather we cannot get anything if we does not select anything. Designing electronic circuits with EHW, after we choose evolving medium, such as Field Programmable Gate Array (FPGA), the circuit which we want to get is constrained to the range of what the medium can do.

As the result, this "Unconstrained" approach on E-HW means that we can extend design space by EAs through removing conventional design method and human biases. Since the part that EAs can be adopted on the current technology is limited, we should make

EAs explore requiring circuits within it, as freely as possible by removing human biases. Moreover, without preventing EAs from this free exploration, we wish to make EAs explore effectively.

We need to consider the details that we have constrained to explore requiring circuit in the design space. [1] indicates that: (1) the bias of evolutionary search makes minimal – but not negligible – assumptions as to the nature of the circuit itself, and (2) they use very direct "genetic" encodings for FPGA, because the circuit representation is one of biases.

We can generally mention some biases on EHW, especially using Genetic Algorithm. That is : (1)The way of evaluating individuals, (2)Genetic Encoding or Genetic Representation, (3)Genetic operators, and (4)Evolving Medium. Applying EHW to autonomous mobile robots, we have to consider more constraints: (5)Interface between FPGA and other parts, (6)Environment for the Robot, and (7)Evaluation and the Robot.

In this paper, we suggest a new genetic encoding method which is not constrained to some biases like the above, specially taking notice of genetic representation, on EHW. The next section shows it, and Section 3 demonstrates an example how to apply it to autonomous mobile robots.

2   Evolving Medium dependent Genetic Encoding

In this section, Evolving Medium dependent Genetic Encoding(EMdGE) is suggested. Fig.1 shows an example of EMdGE.

The left figure (a) in Fig.1 is an example of EMdGE, and the right one (b) is the result that (a) is decoded to FPGA, as an example XC6216 made by Xilinx Inc.

EMdGE describes that branches express cell routing (how cells are connected each other) from each

(a) Chromosome encoded by Evolved Medium dependent Genetic Encoding

(b). FPGA inside configured from the left EMdGE

Fig. 1: Evolving Medium dependent Genetic Encoding

output of the system, and each node expresses data of Cell Functions. As the result, this becomes a tree structure naturally.

The following paragraphs explain the details using the example of Fig.1. Here, we call "Active" to indicate current notable node in EMdGE and current notable cell in FPGA.

Firstly, on the EMdGE(Fig.1(a)), we start from the node Ⓐ which is the current notable node, Active N-ode, because it connects to I/O Block(IOB) as target output, and corresponding to the cell Ⓐ in Fig.1(b), that cell is Active Cell. We follow the circuit from it.

This Active Node Ⓐ is just a connection. In this case, it makes the connection from the branch *1 to the IOB. Then the cell function unit is not used, only cell routing is used on FPGA XC6216.

Branch *1 in the EMdGE shows "S", which tells the direction that the cell described by the next node (Ⓑ) sends signal to the current Active Node Ⓐ. In this case, the direction "S(South)" means that the cell Ⓑ is located upper side of the cell Ⓐ to send signals to the South), and the connection is made between them. Then, the "Active Node" is moved to the next node Ⓑ, and "Active Cell" is moved to the next cell Ⓑ.

When Active Node is Ⓑ which describes NOT Gate, Function unit in the Active Cell Ⓑ makes NOT gate, and makes the connections to the branch *1 and from the branch *2.

The branch *2 also shows "S", which also tells that the next cell Ⓒ is the upper side of the cell Ⓑ described by the node Ⓑ. The connection is made between the cell Ⓑ and Ⓒ, and the "Active" moves to the next node Ⓒ and the next cell Ⓒ.

This decoding nodes and branches to Cell Function and Cell routing are continued. If a node has two or three branches as node Ⓒ has 3 branches (in this case, the number of branches that a node can have is up to 3), the left subtree (toward the node Ⓓ) of EMdGE is decoded at first, the middle subtree is the next, and the right subtree is the end.

It stops decoding when the Active Node is NULL Ⓔ, or a routing descibes to be out of the cell array on the target FPGA chip as the node Ⓕ and its sub-tree.

Even though many nodes indicate the same cell in cell array of FPGA, for example, node Ⓐ, Ⓖ and Ⓗ indicate the same cell, they are decoded because they are not conflicting. But, if some nodes conflicts each other, for example, node Ⓘ and Ⓙ indicate the same cell, and these nodes express different Cell Functions and inputs, the later node (now, it is the node Ⓙ) overwrites its cell function and cell routing data to decode the cell Ⓙ, to the cell and its function unit where the previous one Ⓘ is decoded.

By the way, how can this EMdGE method operate external input signals from FPGA outside?

Fig.2 shows the inside of FPGA, (a) indicating be-

(a). FPGA Turret Inside (before configuring an Individual)

(b). Example of an Individual

Fig. 2: FPGA Inside : (a) shows cell routing across whole of cell array in order to spreading input signals, before decoding an EMdGE code. (b) shows FPGA inside after decoding an EMdGE, which comes into contact with input signal routing.

Rigorously, our experiment with FPGA Turret [4] (made by Applied AI Systems, Inc.) which mount Xilinx XC6216 FPGA chip, mounted on Miniature Autonomous Mobile Robot Khepera[2]. This figure show FPGA XC6216 inside dependent on this FPGA Turret with respect to the fixed input and output positions.

fore decoding of an EMdGE, and (b) after that.

Many routing (seen in Fig.2(a)) are made before EMdGEs are decoded to make input signals go and come across FPGA Cell Array. Since these routing may be overlapped by EMdGE decoded data, input signals may connect to cells configured by EMdGE decoding, that is, the phenotype EMdGE can receive external input signal. Fig.2(b) presents such a case.

This EMdGE is expected for some benetifs.

Firstly, electronic circuits usually make the chain of electronic elements. EMdGE pays attention to this thing. This gives EAs the design space as freely as possible within the target medium, and removes the design space that expresses individuals impossible to make chain of electronic elements.

Secondly, this EMdGE is not restricted to the number of cells and to the area of evolving cells, only with FPGA chip. In the case of the experiment which uses FPGA in [1], it is restricted to only the 10 × 10 cell array in FPGA. These 100 cells they guessed were enough to make circuit for descriminating 2 types of square signals. But, in practice, how can we guess the size of cell array used on EAs? Using EMdGE, we do

not need to guess it.

Lastly, we can define the rule of EMdGE encoding adjusted to the structure of the target medium in which we want to evolve, such as FPGA. This can decide the restriction of variation of the target medium, in other words, this becomes one of deciding the extent of design space in the target medium. We have to define carefully in order to become the maximum of the rate of requiring circuit space in the design space within the target medium.

3 An Experiment

Using this EMdGE, we have evolved autonomous mobile robot controller within FPGA to make its intellectual behavior without CPU and its program.

We use a miniature mobile robot "Khepera" [2], and Simple Genetic Algorithm (GA) [3] with EMdGE. Khepera mounts FPGA Turret made by Applied AI Systems, Inc. [4], which has an FPGA XC6216 made by Xilinx Inc. [5].

Sensor inputs are connected to FPGA, and FPGA output is connected to Motor Control through Khepera BIOS, without other operations in CPU. That is, Khepera runs as follows:

1. Khepera BIOS gets the values of each Infra-Red Sensor input.

2. 8 Sensor values are operated for a threshold, and exchange them into 8-bit input value for FPGA.

3. The value inputs to FPGA.

4. FPGA configured from an EMdGE choromosome outputs 2-bit signal.

5. Khepera samples 2-bit output signal, and it decides whether each motor is accelerated or decelerated.

6. Khepera controls motors with new speeds.

These 1-6 operations are repeated. If FPGA has effective circuit, Khepera may run intellectually.

Operation 5 is that Khepera samples FPGA outputs 2 times per 1 milli-second on each bit corresponding to each motor. If 2 values for each motor are different, then the motor is accelerated, otherwise the motor is decelerated. This means that FPGA can run motors if FPGA forms a kind of oscillator, and this method can express rich behaivor of the robot.

To run Khepera intellectually, we applied GA with EMdGE to make effective circuit in FPGA, as intellectuall controller of Khepera. We only change the

structure of chromosome of GA. But, using EMdGE in Simple GA, Crossover changes into that randomly selected nodes in two selected individuals respectively as parents exchange each other, and Mutation also changes into that a randomly selected node and its subtree in each child individual replaces new subtree randomly generated. Elite strategy for selecting a few offsprings, and roulette wheel strategy for selecting parents are used.

On the evaluation step, after FPGA equipped on Khepera is configured, it make Khepera run freely controled by FPGA, in the $50cm \times 38cm$ rectangular area, with no walls inside, and walls are only on the border of the arena.

Our experiments are still continued now. Since Khepera operation 5 as stated above are used, Khepera can move only if it has a kind of oscillator circuit. Therefore we give an individual in the initial population to be able to make such circuit.

Khepera has run and evolved to run actively more and more. We still continue to make this experiment with FPGA on EAs, and report it in the presentation.

4 Related works

We discuss comparing our suggested method and others in this section.

In the experiment by A. Thompson et al. [1], concentrating on "Unconstrained" approach without rapid evolving, binary bitstring directly used at FPGA XC6216 configuration are adopted as gene to GA.

We recognize it, and EMdGE is also based on "Unconstrained" approach, furthermore, the size of genes is unlimited, and is considered the effectiveness to evolve, although it may have other constraint. We are not enough in this paper to prove how much this method is free and effective. And we need to carry out such experiments and compare them more.

EMdGE forms Tree Structure consequentially. Remembering Genetic Programming (GP) from what considers chromosome forming tree structure, EMdGE, however, is different from it. [6] written by J. R. Koza et al. collects their works on GP adopted to automatically design analog circuits. The method using GP is based on Programming so much, in addition to influencing that individuals are evaluated on simulation. Each circuit is described by the circuitconstructing program trees, altering circuit topology, inserting components into the circuit, appearing in subtrees as arguments, reusing them and so on.

On the other hand, our EMdGE is based on the evolving medium at first, and the way of encoding is made depended on the structure of the medium, even though chromosomes of EMdGE also form tree structure only consequentially. GP and its results, however, may lead EMdGE to make some circuits evolve better, then we may learn from it.

5 Summary

We reconsidered Unconstrained Intrinsic Evolvable Hardware, and try to extend the effectiveness exploring electronic circuits by suggesting a new genetic encoding. That is Evolving Medium dependent Genetic Encoding (EMdGE) which can try to raise the rate of requiring circuit space in the whole of design space.

We are continuing this experiments, and are considering how to adopt EHW for intellectuall autonomous mobile robots.

Acknowledgement

References

[1] A. Thompson, P. Layzell, and R. S. Zebulum, "Explorations in Design Space: Unconventional electronics design through artificial evolution", *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 3, pp167-196, 1999.

[2] F. Mondada, E. Franzi and P. Ienne, "Mobile robot miniaturization: A tool for investigation in control algorithms", T. Yoshikawa and F. Miyazaki, (Eds.), *Proceeding of the Third International Symposium on Experimental Robotics 1993*, Springer Verlag, 1994, pp501-513, Kyoto, Japan

[3] D. E. Goldberg: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Massachusetts, USA, 1989

[4] Applied AI Systems, Inc., *Quick Start FPGA Turret manual*, 1999

[5] Xilinx, *XC6200 Field Programmable Gate Arrays manual*, Version 1.10, 24, 1997

[6] J. R. Koza, F. H. Bennett III, D. Andre, and M. A. Keane, *Genetic Programming III*, Morgan Kaufmann Publishers Inc., 1999.

Obstacle Avoidance for Autonomous Mobile Robot
Using Anytime Sensing

K. Fujisawa[*1]
T. Suzuki[*1]

S. Hayakawa[*2]
S.Okuma[*1]

T. Aoki[*3]

[*1]Electrical Eng.,
Nagoya University
Furo-cho,Chikusa-ku,Nagoya,
Aichi,464-8603,Japan

[*2]Toyota Technological
Institute
2-12, Hisakata, Tenpaku-ku,
Nagoya, 468, Japan

[*3]Nagoya Municipal
Ind., Res., Inst.
3-4-41, 6-ban, Atsuta-ku,
Nagoya, 456, Japan

Abstract

We propose a new method for moving obstacles avoidance for an autonomous mobile robot. In the proposed method, a robot decides the calculation time required for an action search, and acquires the suitable motion in real time. Framework of the Anytime Sensing is also adopted and expanded, which increases a sensing information required for the action search according to increase of the search time. With this property, it could be possible for the robot to search one of the 'good' motion in less time because the calculation load increases according to increase of the search time and, as the result, we can make the search process faster by introducing anytime sensing. In this paper, we explain our proposed method, and show the usefulness through some simulation results.

Keywords: Moving Obstacle Avoidance, Real-Time Search, Anytime Sensing, Autonomous Mobile Robot

1 Introduction

Recently, many methods have been proposed for moving obstacles avoidance for an autonomous mobile robot. In those methods, an autonomous mobile robot must know the pre-defined rules for motions[1], or learn the motion rules through the interaction with the environment when it moves in the dynamic environment[2]. Most of researches in this field have used one of them or a hybrid type[3]. In the rule-based method, the pre-defined knowledge of the environment is required in order to design the motion rules. But it is not straightforward to know about the environment in advance, and the great efforts are often required for the designer. On the other hand, in the reinforcement learning scheme, it is necessary to design the efficient categorization of the environment to make the learning process feasible. Moreover,

it needs a lot of experiments to acquire the various rules. To overcome these problems, we have proposed a new motion planning method for moving obstacles avoidance problem[4]. In our previous work, the robot can avoid obstacles and go toward the goal by recognizing the current situation, searching and executing fine motions in real time. This method was also based on the framework of Anytime Algorithm [5][6] which improves a quality of search result over the calculation time and is an interruptible algorithm at anytime. The crucial point of this method is to be able to balance the amount of calculation time and the quality of the acquired motion. For example, in the safe situation, the robot has enough time to search the optimal motion. On the contrary, the robot has to search a desirable motion in very short time in the dangerous situation even if the result does not have good quality. Anytime Algorithm can balance such a simultaneous requirement as mentioned above and has been expanded so as to accommodate with the dynamic environment in [4].

When the robot tries to acquire the good motion in real time, it would be necessary to consider that the action search should be done with limited computational power. Generally speaking, if the robot has enough information on the environment, it would become easier for the robot to acquire the optimal action. However, this results in a vast computational amount and, as the worst case scenario, infeasibility in real time execution. Thus, the more the sensory information of the environment is utilized for the search, the more time it takes to process it. Therefore, it is necessary to consider the trade-off between the quality of the information of the environment and computational amount required for search. Besides, this trade-off becomes more significant as the environment becomes complicated. In such a case, it is desirable to accel-

erate the search process by gradually increasing the quality of the information required for action search according to increase of the time.

To realize this idea, we introduce the Anytime Sensing[7] which can balance the trade-off between the quality of the information and the calculation time. Since the strategy introduced in [7] deals with only static environment, we expand the framework in [7] to the search problem in the dynamic environment, and then we realize the real time motion planning system for moving obstacle avoidance for an autonomous mobile robot.

Section 2 describes the introduction and expansion of anytime sensing. In Section 3, we explain the motion planning system for moving obstacles avoidance. In Section 4, we show the usefulness of this method with some simulation results. Finally, we conclude this paper.

2 Introduction and Expansion of Anytime Sensing

In this research, we adopt Anytime Sensing algorithm for the real-time motion planning to avoid moving obstacles. This algorithm, however, is not directly used in the dynamic environment because the algorithm has been developed in the static environment. The environment around the robot is always changing. The allocated time to search should be flexible according to the situation around the robot. When the robot is in the safe situation, it can search a fine action in enough computational time with the fine(high-quality) sensory information. On the contrary, when the robot is in the dangerous situation, it should quickly decide a certain action using the rough(low-quality) sensory information. The allocation of the computational time can not be designed in advance. Moreover, the action search process may be interrupted before convergence because an unpredictable situation could occurs. According to this, we expand Anytime Sensing to apply to the search in the dynamic environment. In this research, the quality of the information used for the search gradually increases over the calculation time. That is, the search is started using the environmental information whose quality is the lowest. Then, the quality gradually increases as the search goes on. According to this, the robot can acquire a certain motion in a short time even if the robot is in the dangerous situation or the search is interrupted because of the unpredictable situation. Moreover, the search can converge smoothly due to use the information whose quality become higher and higher.

3 Motion Planning System for Moving Obstacles Avoidance

3.1 Framework of Problem

Many methods for the obstacle avoidance have been proposed under the diverse conditions to the problem. In this paper, the following conditions are assumed.

1. The robot avoids obstacles with both steering and velocity control inputs simultaneously.
2. There are only moving obstacles in the environment around the robot.
3. The robot with two wheels can move to all the directions without path limit.
4. The goal of the robot is given in advance.
5. The robot and obstacles move smoothly.
6. The obstacles move straightly at the same speed.

Under these conditions, the motion planning system to avoid moving obstacles is constructed.

3.2 Whole system

Figure 1 illustrates the motion planning system for the moving obstacles avoidance. The flow of the system is as follows.
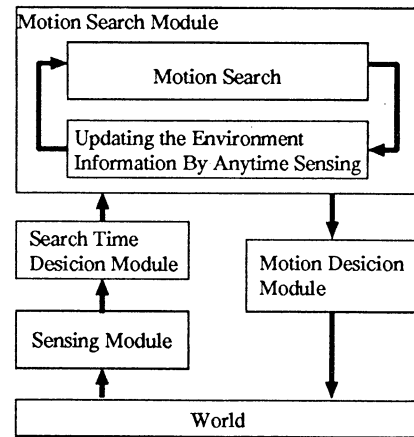
Figure 1: Motion Planning System

1. In the sensing module, the robot catches the information around the robot and predicts the situation where the robot will execute a motion.
2. In the search time decision module, the time to search the next motion is decided.
3. In the motion search module, the suitable motion for the predicted situation is searched using the information of the environment given by Anytime Sensing.
4. After the search, the control inputs to the robot are calculated, and the next motion is decided.

The details of this flow are described in the following subsections.

3.3 Decision of Search Time

In this system, the robot predicts the situation where it will execute a motion, and then it searches the suitable motion for the predicted situation only when it is necessary to avoid obstacles. In there is no need to avoid obstacles, the robot moves to the goal until it reaches the goal. Moreover, the search time is variable depending on the situation. If obstacles move near the robot, the robot searches a motion in a short time. On the other hand, the robot searches the motion until the distance between the robot and obstacles become close. Figure 2 shows the relative coordinates where the robot is the origin. LA means the distance from the robot to the predicted cross point of the moving direction of the robot and an obstacle. The shorter LA is, the more dangerous the obstacle is for the robot. When LA is shorter than the sum of the radius of both the robot and an obstacle, the robot collides the obstacle. According to this, all LA are calculated to check wheter the robot has to avoid obstalces. If LA is shorter than a certain threshold, it is regarded that it is necessary for the robot to avoid the obstacle. In such a case, the search time is a half of the time until the obstacle rearches the predicted cross point, and then the robot searches the fine motion in the decided time. If there are more than two obstacles that it is necessary for the robot to avoid, the search time is the smallest one.
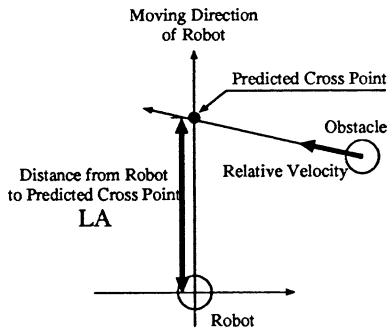
Figure 2: Decision of the Search Time

3.4 Application of Anytime Sensing

As mentioned in the previous section, the robot searches a motion using the information of the environment whose quality gradually increases over the calculation time. Figure 3 illustrates the method that Anytime Sensing updates the information about the environment in this system. The circle which contains all obstacles is made like Figure 3(a) before starting search. We call this circle the obstacle existence area.

In the motion search, this area is regarded as an obstacle. This representation is the information of the environment whose quality is the lowest given by Anytime Sensing. The robot searches the motion that the robot can avoid this area. Then, this area is divided into two areas like Figure 3(c) as the search process goes on. The way to divide areas is follows. This area are divided into the right and left side by the vector from the robot to the center of the circle like Figure 3(b). Then, the new two areas are calculated based on the information of obstacles in each area like Figure 3(c). When there are more than two obstacle existence areas, the area which includes the most number of obstacles is divide.
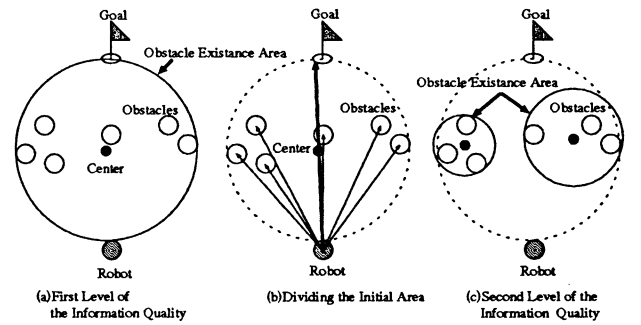
Figure 3: Application of Anytime Sensing

3.5 Action Search

In this system, the motion of the robot is decided by the velocity and steering control. Because of this, the robot searches a pair of these control inputs. In this research, we adopt Evolution Strategy(ES) as a search method. The evaluation function is shown as the Equation(1).

$$Evaluation = \alpha \cdot G + (1 - \alpha) \cdot V \qquad (1)$$

α : weight $(0 \le \alpha \le 0.5)$
G : the evaluation value for the goal
V : the evaluation value for the obstacle avoidance

The parameter α is propotion to the distance from the robot to the nearest obstacle. That is, the evaluation function is the avoidance-oriented when the obstacles are near from the robot. On the contrary, when the obstacles are far from the robot, this is the goal-oriented evaluation function. The values of G and V are respectively calculated as follows.

$$V = \frac{1}{t_2} \int_{l_1}^{l_2} S \cdot W(L, RA) dl \qquad (2)$$

$$W = \frac{1}{(1 + exp(\frac{RA_{th} - |RA|}{c_1}))} \cdot \frac{1}{(1 + exp(\frac{L_{th} - L}{c_2}))} \quad (3)$$

l_1, l_2	starting and ending point of the trajectory of the obstacle area's center
S	the obstacle existence area
t_2	search time
W	density in the space
L	relative distance from the robot to an obstacle
RA	relative angle from the robot to an obstacle
$RA_{th}, c_1, L_{th}, c_2$	the design parameters

Equation (2) means the volume of an oblique column with the density defined by Equation (3) as Figure 4 shows. Figure 4 illustrates the 3-dimensional space where it is the relative coordinate where the robot is the origin. In this space, the trajectories of the robot and obstacle existence area are drawn as oblique columns. The smaller the distance from the robot to the robot and the relative angle to the obstacle are, the heavier density in the space is. The smaller this volume is, the safer motion the robot selects for the obstacle avoidance. The volume of the obstacle existence area is calculated using the trajectory of the center and making the obstacle existence area constant this time, and it is regards as the evaluation value for avoidance. When the trajectories of the obstacle existence areas cross over the one of the robot , the volumes become extremely high.
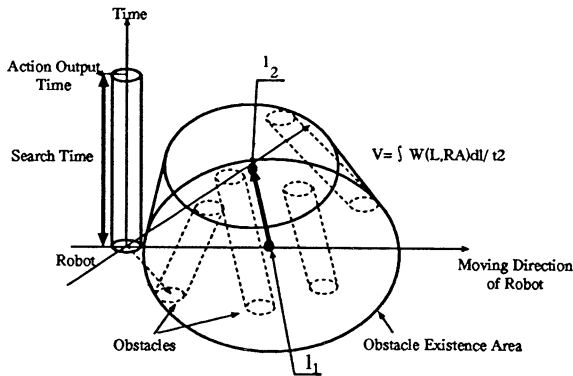
Figure 4: Evaluation Value for the Obstacle Avoidance

On the other hand, the evaluation value for the goal is calculated by Equation (4). The more the robot goes to the goal, the smaller this value becomes and the better motion the robot selects for reaching the goal.

$$G = \frac{1}{1 + exp(\frac{\frac{\Delta RTL}{t_2} - V_{th}}{c_3})} \quad (4)$$

ΔRTL	tha change of the distance to goal
V_{th}, c_3	the design parameters

4 Simulation Results and Verification

In this section, we show the simulation results and the usefulness of our method.

4.1 Simulation Results

Let the following be assumed in the simulation.

1. The number of obstacles is five.
2. The positions and motions of obstacles are set at random.
3. The obstacles move straight at the same speed.
4. The goal position is given in advance.
5. The robot motion and the search time at the first step are decided in advance.
6. The information given by Anytime Sensing is updated every ten generations in ES.
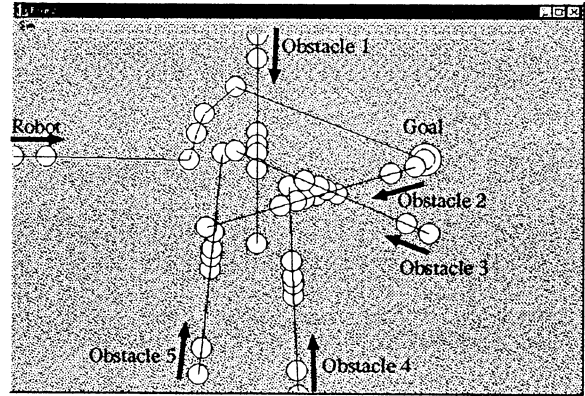
Figure 5: Simulation Result(No.1)

Figure 5 and Figure 6 illustrate the simulation results. The white circles in each figure indicate the positions of the robot and obstacles at that time when the robot executed motions: i.e. The robot searched and decided its own motion while moving to next circle. The time to move from the position of a certain circle to the next one is the search time, and it is decided by the robot.

Figure 5 shows the case where the obstacles approach the robot from right, left and front side. In the
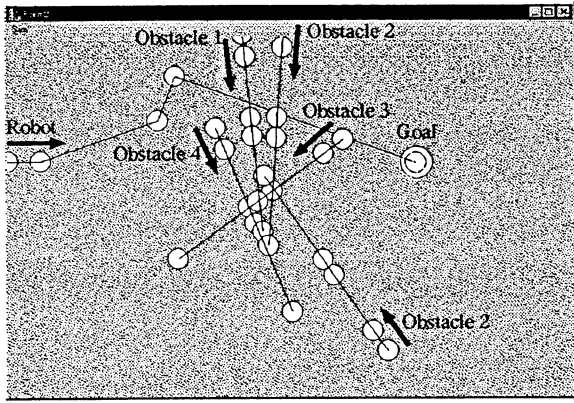
Figure 6: Simulation Result(No.2)

Figure 7: Evaluation Value for the Obstacle Avoidance(without Anytime Sensing)

beginning of the trial, the robot decided to have the long search time because all obstacles were far from the robot, and it searched a fine motion using the high-quality information of the environment according to Anytime Sensing. As a result, the robot moved straightly, and at the next step it began to avoid obstacles using the large steering to the left. After avoiding all obstacles, the robot moved to the goal straightly because there was no obstacle.

Figure 6 shows the case where the four of all obstacles approached the robot from the left side and the other approached from the light side. In this case, the robot turned to the left at the second and third step to avoid obstacles, then it moved to the goal straightly.

According to these result, it is found that the robot can acquire the proper motions by recognizing the current situation, deciding the search time and searching the fine motion in real time. Moreover, the robot can search the fine motion using the information of the environment which quality is gradually increasing.

4.2 Verification

In this subsection, we show the verification of our method using some simulation data which were acquired from two simulation results shown in the previous subsection. We performed the simulations ten times in each case. As a comparison, we carried out the simulation in the case where Anytime Sensing was not adopted into the motion planning system. In this case, the volumn obtained by Equation (2) was respectively calculated for each obstacles as Figure 7 shows. The evaluation value for the obstacles avoidance V were the average volumn.

Figure 8 and Figure 9 show the histories of the evaluation values for the search at the only first step in
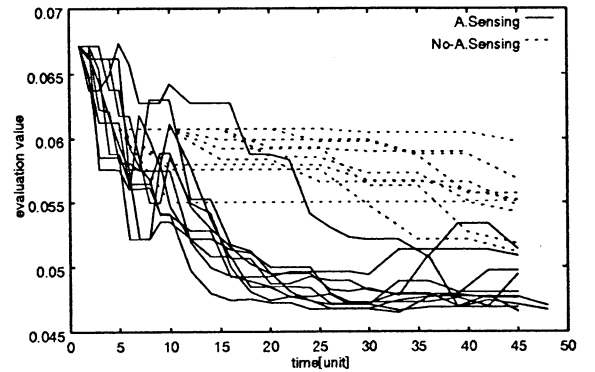
Figure 8: Succession of the Evaluation Value(Sim1)

the case of the Figure 5 and Figure 6 respectively. This is because the evaluation values of motions after the first step are influenced by the first step result. The horizontal axis indicates the time. One time unit means the time to calculate a volumn explained by Equation(2). The longitudinal axis indicates the evaluation value shown by Equation (1). In each figure, the thick line indicates the evaluation value in the case using Anytime Sensing, and the dotted line indicates one in the case where Anytime Sensing is not adopted. The evaluation value in the case where Anytime Sensing is applied is different from one in the case without applying Anytime Sensing. This is because the evaluation value used Anytime Sensing is calculated based on the center of the obstacle existence area. Because of this, the evaluation value where Anytime Sensing is applied is converted into one in the case without Anytime Sensing as Figure 7 shows.

In the case of Figure 8, it takes less time to acquire the initial solution in the case using Anytime Sensing than that in the case without Anytime Sensing. This
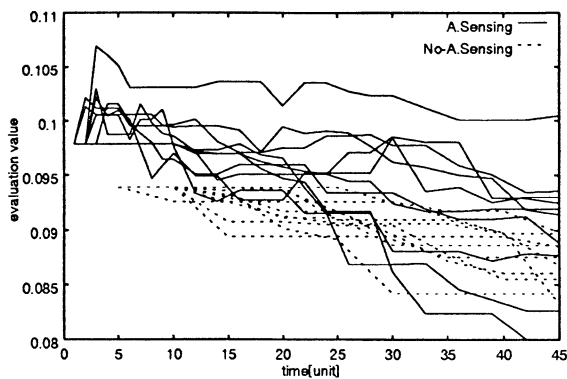
Figure 9: Succession of the Evaluation Value(Sim2)

is because the evaluation value is calculated using the obstacle existence area in the search using Anytime Sensing. In this case, this area is regarded as a single obstacle, and then the evaluation value is calculated to this group. On the other hand, the evaluation value is calculated to all five obstacles when the Anytime Sensing is not adopted. Because of this, the load of the calculation in the case using Anytime Algorithm is low in comparison with the case without Anytime Sensing. Moreover, the search using Anytime Sensing converged faster than that without Anytime Sensing in the most of trials. This is because of the following reasons. At first, the load of the calculation is low as mentioned before. At the beginning of the search, the evaluation value in the case using Anytime Sensing is worse than that in the case without Anytime Sensing. This is because the quality of the information used in the search is low. But, the search can converge faster by repeating the search in the short time using Anytime Sensing.

In the Figure 9, the search using Anytime Sensing did not converge well although it costs less time to acquire the initial solution in the case using Anytime Sensing than that without Anytime Sensing and the search was repeated many times. This is because the information given by Anytime Sensing did not summarize the real situation of the obstacles. In the case of Figure 9, the area did not summarize the real situation of obstacles because the real position of obstacles is biased in the initial area.

According to these results, the robot can acquire a certain solution faster in the case where Anytime Sensing is applied, because the calculation load become low. Moreover, it is found that the representation of the information by Anytime Sensing accelerates the search convergence when it summarize the real situation of obstacles well. In this research, only

one method is adopted to apply Anytime Sensing. For the future works, we will study the case where it is possible to apply Anytime Sensing and the combination of the various methods to apply Anytime Sensing as needed.

5 Conclusions

We propose a new method for the moving obstacles avoidance using Anytime Sensing. In our proposed method, the robot can acquire the fine motion in the short time by decreasing the quality of the information about environment. Moreover, the search can converge smoothly because the quality of the information increases gradually. For the future works, we will consider the method to combine the various method to apply Anytime Sensing. Moreover, we will adopt learning system in the proposed system. To memorize the calculated motion, the robot use the computational resources, efficiently.

References

[1] Y.Maeda et.al: "Avoidance Control among Moving Other robots for a Mobile Robot on the Fuzzy Reasoning",J.of the Robotics Society of Japan,Vol.6,No.6,pp.518-522,1988

[2] T.Tsubouchi et.al: "A navigation scheme with learning for a mobile robot among multiple moving obstacle",1993 IEEE International Conference on Intelligent Robots and Systems, Yokohama in Japan, July26-30,pp.2234-2240

[3] T.Aoki et.al: "Acquisition of Optimal Action Selection to Avoid Moving Obstacles in Autonomous Mobile Robot", Proc. of IEEE International Conference on Robotics and Automation,Minneapolice in U.S.A,Apr.22-28, pp.2055-2060, 1996

[4] K.Fujisawa et al.: "Real Time Motion Planning for Autonomous Mobile Robot using Framework of Anytime Algorithm",Proc. of IEEE International Conference on Robotics and Automation, Detroit in U.S.A, May 10-15, pp.1347-1352, 1999

[5] S.Zilberstein: "Using Anytime Algorithms in Intelligent Systems", AI Magazine,17(3),pp.73-83,1996

[6] J.Grass: "Reasoning about Computational Resource Allocation -An introduction to anytime algorithms",the ACM magazine,1996

[7] S.Zilberstein et al.: "Anytime Sensing, Planning and Action: A Practical Model for Robot Control", the ACM magazine,1996

[8] Ralf Salomon: "Increasing Adaptivity through Evolution Strategies", The Fourth International Conference on Simulation of Adaptive Behavior,1996

Author Index