PROCEEDINGS OF THE NINTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 9th '04)

Vol. 1

Jan. 28-Jan. 30, 2004
B-Con Plaza, Beppu, Oita, JAPAN

Supported by · Ministry of Education, Culture, Sports, Science and Technology, Japanese Government (MEXT)
· Air Force Office of Scientific Research, Asian Office of Aerospace Research and Development (AFOSR/AOARD)

Editors : Masanori Sugisaka and Hiroshi Tanaka
ISBN4-9900462-4-2

Proceedings of The Ninth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 9th '04)

January 28-30, 2004
B-Con Plaza, Beppu, Oita, Japan

Editors: Masanori Sugisaka and Hiroshi Tanaka

THE NINTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 9th '04)

ORGANIZED BY

Organizing Committee of International Symposium on Artificial Life and Robotics (Department of Electrical and Electronic Engineering, Oita University, Japan)

CO-SPONSORED BY

Santa Fe Institute (SFI, USA)
The Society of Instrument and Control Engineers (SICE, Japan)
The Robotics Society of Japan (RSJ, Japan)
The Institute of Electrical Engineers of Japan (IEEJ, Japan)
Institute of Control, Automation and Systems Engineers (ICASE, Korea)
Chinese Association for Artificial Intelligence (CAAI, China)

COOPERATED BY

The Institute of Systems, Control and Information Engineers (ISCIE)
The Institute of Electronics, Information and Communication Engineers (IEICE)
The Institute of Electrical and Electronics Engineers, Inc.(IEEE Japan Council)
Japan Robot Association (JARA)

SUPPORTED BY

Ministry of Education, Culture, Sports, Science and Technology, Japanese Government (MEXT)
Air Force Office of Scientific Research, Asian Office of Aerospace Research and Development (AFOSR/AOARD)
ISAHAYA ELECTRONICS CORPORATION
MITSUBISHI ELECTRIC CORPORATION ADVANCED TECHNOLOGY R&D CENTER
OITA GAS CO., LTD.
STK TECHNOLOGY CO., LTD.
SANWA SHURUI Co., Ltd.
YATSUSHIKA BREWERY CO., LTD.
Kyushu Bureau of Economy, Trade and Industry, Ministry of Economy, Trade and Industry
OITA PREFECTURE
OITA CITY
Beppu City
The Federation of the Chamber of Commerce and Industry in Oita Prefecture
The Oita Prefectural Organization of the Industrial Groups

KYODO NEWS
JIJI PRESS
OITA GODO SHIMBUNSYA
The Asahi shimbun Company
The Mainichi Newspapers Co.
THE YOMIURI SHIMBUN
The Nishinippon Newspaper Co.
THE NIKKAN KOGYO SHIMBUN, LTD.
Nihon Keizai Shimbun, INC.,
NHK Oita Station
Oita Broadcasting System
Television Oita System
Oita Asahi Broadcasting Co. Ltd.
Oita System Control Research Society

ADVISORY COMMITTE CHAIRMAN

F.Harashima (Tokyo Denki University, Japan)

GENERAL CHAIRMAN

M.Sugisaka (Oita University, Japan)

CO-GENERAL CHAIRMAN (PROGRAM)

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

CO-CHAIRMAN

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

ADVISORY COMMITTEE

F.Harashima (Tokyo Denki University, Japan)
H.Kimura (The University of Tokyo, Japan)
T.Fukuda (Nagoya University, Japan)
S.Ueno (Kyoto University, Japan)
M.Tomizuka(University of California at Berkeley, USA)

INTERNATIONAL ORGANIZING COMMITTEE

K.Aihara (The University of Tokyo, Japan)
W.B.Arthur (Santa Fe Institute, USA)
W.Banzhaf (University of Dortmund, Germany)
C.Barrett (Los Alamos National Laboratory, USA)
M.Bedau (Reed College, USA)
Z.Bubnicki (Wroclaw University of Technology, Poland)
J.L.Casti (Santa Fe Institute, USA)
H.S.Cho (KAIST, Korea)
J.M.Epstein (The Brookings Institution, USA)
T.Fukuda (Nagoya University, Japan)
H.Hashimoto (The University of Tokyo, Japan)

D.J.G.James (Coventry University, UK)
S.Kauffman (Santa Fe Institute, USA)
C.G.Langton (Santa Fe Institute, USA)
M.H.Lee (Pusan National University, Korea)
J.J.Lee (KAIST, Korea)
M.Nakamura (Saga University, Japan)
G.Obinata (Nagoya 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)
G.Wang (Tsinghua University, China)
W.R.Wells (University of Nevada-Las Vegas, USA)
Y.G.Zhang (Academia Sinica, China)

INTERNATIONAL PROGRAM COMMITTEE

K.Abe (Tohoku University, Japan)
K.Aihara (The University of Tokyo, Japan) (Co-chairman)
T.Arita (Nagoya University, Japan)
H.Asama (The University of Tokyo, Japan)
M.Bedau (Reed College, USA)
R.Belew (University of California-San Diego, USA)
Z.Bubnicki (Wroclaw University of Technology, Poland)
T.Christaller (Fraunhofer Institute for Autonomous intelligent Systems-AiS)
C.S.Han (Hanyang University, Korea)
I.Harvey (University of Sussex, UK)
H.Hashimoto (The University of Tokyo, Japan)(Co-chairman)
K.Hirasawa (Waseda University, Japan)
H.Hirayama (Asahikawa Medical College, Japan)
N.Honma (Tohoku University, Japan)
T.Ikegami (The University of Tokyo, Japan)
H.Inooka (Tohoku University, Japan)
K.Ito (Tokyo Institute of Technology, Japan)
J.Johnson (The Open University, UK)
Y.Kakazu (Hokkaido University, Japan)
O.Katai (Kyoto University, Japan)
S.Kawaji (Kumamoto University, Japan)
S.Kawata (Tokyo Metropolitan University, Japan)
S.Kitamura (Kobe University, Japan)
T.Kitazoe (University of Miyazaki, Japan)
S.Kumagai (Osaka University, Japan)
J.M.Lee (Pusan National University, Korea)
C.G.Looney (University of Nevada-Reno, USA)
H.H.Lund (University of Southern Denmark, Denmark)
M.Nakamura (Saga University, Japan)
R.Nakatsu (KWANSEI GAKUIN University, Japan)
S.Omatu (Osaka Prefecture University, Japan)

M.Okamoto (Kyushu University, Japan)
R.Pfeifer (University of Zurich-Irchel, Switzerland)
T.S.Ray (University of Oklahoma, USA) (Co- chairman)
T.Sawaragi (Kyoto University, Japan)
T.Shibata (AIST, Japan)
K.Shimohara (ATR, Japan)
M.Sugisaka (Oita University, Japan)
H.Tanaka (Tokyo Medical & Dental University, Japan)
(Chairman)
M.Tanaka-Yamawaki (Tottori University, Japan)
N.Tosa (ATR, Japan)
K.Ueda (The University of Tokyo, Japan)
K.Uosaki (Osaka University, Japan)
H.Wakamatsu (Tokyo Medical & Dental University, Japan)
K.Watanabe (Saga University, Japan)
M.Yano (Tohoku University, Japan)

LOCAL ARRANGEMENT COMMITTEE

X.Feng (University of Singapore, Singapore)
A.Loukianov (Baikal State University of Economy & Law, Russia)
M.Rizon (Northern Malaysia University College of Engineering, Malaysia)
K.Shibata (Oita University, Japan)
M.Sugisaka (Oita University, Japan) (Chairman)
Y.Suzuki (Tokyo Medical & Dental University, Japan)
J.Wang (Oita University, Japan)
X.Wang (Oita Institute of Technology, Japan)

HISTORY

This symposium was founded in 1996 by the support of Science and International Affairs Bureau, Ministry of Education, Culture, Sports Science and Technology, Japanese Government. Since then, this symposium has been held every year at B-Con Plaza, Beppu, Oita, Japan except in Oita, Japan (AROB 5th '00) and in Tokyo, Japan (AROB 6th '01). The ninth symposium will be held on 28-30 January, 2004, at B-Con Plaza, Beppu, Oita, Japan. This symposium invites you all to discuss development of new technologies concerning Artificial Life and Robotics based on simulation and hardware in the twenty first century.

OBJECTIVE

The objective of this symposium is the development of new technologies for artificial life and robotics which have been recently born in Japan and are expected to be applied in various fields. This symposium will discuss new results in the field of artificial life and robotics.

TOPICS

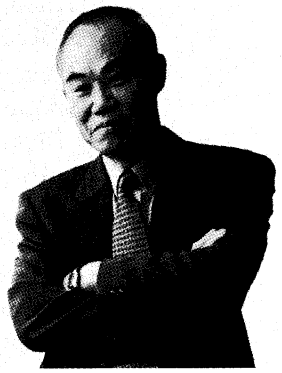
Artificial brain research
Artificial intelligence
Artificial life
Artificial living
Artificial mind research
Bioinformatics chaos
Brain science
Cognitive science evolutionary computations
Complexity
Computer graphics
DNA computing
Fuzzy control
Genetic algorithms
Human-machine cooperative systems
Human-welfare robotics
Innovative computations
Intelligent control and modeling
Micromachines
Micro-robot world cup soccer tournament
Mobile vehicles
Molecular biology
Multi-agent systems
Nano-biology
Nano-robotics
Neural networks
Neurocomputers
Neurocomputing technologies and its application for hardware
Pattern recognition
Robotics
Robust virtual engineering
Virtual reality

COPYRIGHTS

Accepted papers will be published in the proceeding of AROB and some of high quality papers in the proceeding will be requested to re-submit fir the consideration o publication in an international journal ARTIFICIAL LIFE AND ROBOTICS (Springer) and APPLIED MATHEMATICS AND COMPUTATION (North-Holland). All correspondence related to the symposium should be addressed to AROB Secretariat

Dept. of Electrical and Electronic Engineering,
Oita University
700 Dannoharu, Oita 870-1192, JAPAN
TEL : +81-97-554-7841, FAX : +81-97-554-7818
E-MAIL arobsecr@cc.oita-u.ac.jp
Home Page http://arob.cc.oita-u.ac.jp/

MESSAGE



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

It is my great honor to invite you all to the upcoming International Symposium on Artificial Life and Robotics. The first symposium was held in February (18-20) 1996, B-Con Plaza, Beppu, Oita, Japan. That symposium was organized by Oita University under the sponsorship of the Japanese Ministry of Education, Science, Sports, and Culture (Monbusho), and co-sponsored by Santa Fe Institute (USA), SICE, RSJ, and IEEJ, (Japan). I would like to express my sincere thanks to the Science and Technology Policy Bureau, Ministry of Education, Culture, Sports, Science and Technology (Monkasho), Japanese Government, for their repeated support.
This symposium is supported by Monkasho, Air Force Office of Scientific Research, Asian Office of Aerospace Research and Development (AFOSR/AOARD) and other institutions. The symposium invites you to discuss the development of new technologies in the 21st century, concerning Artificial Life and Robotics, based on simulation and hardware.

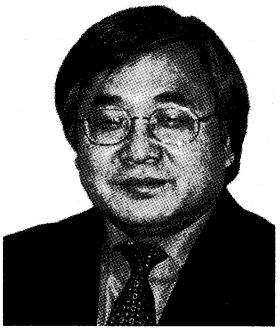
We hope that AROB will facilitate the establishment of an international joint research institute on Artificial Life and Robotics. I hope that you will obtain fruitful results from the exchange of ideas during the symposium.

Masanori Sugisaka
M. Sugisaka

December 20, 2003

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 Ninth International Symposium on Artificial Life and Robotics (AROB 9th '04). This symposium is made possible owing to the cooperation of Oita University and Santa Fe Institute. We are also debt to Japanese academic associations such as SICE, RSJ, and several private companies. I would like to express my sincere thanks to all of those who make this symposium possible.

As is needless to say, the complex systems or Alife approach now attracts wide interests as a new paradigm of science and engineering. Take an example in the field of bioscience. The accomplishment of HGP (Human Genome Project) has published the special issue of Nature, and HGP (Human Genome Project), vast amount of genome information brings about not only from human genome but also various species like several bacterias, yeast, warm, fly. However, as a plenty of genome data becomes available, it becomes sincerely recognized that the framework by which these genome data can be understood to make a whole picture of life is critically need thus, in the "post genomic era , the complex systems or Alife approach is now actually expected to be an efficient methodology to integrate this vast amount of data.

This example shows the complex system approach is very promising and becomes widely accepted as a paradigm of next generation of science and engineering. We hope this symposium becomes a forum for exchange of the ideas of the attendants from various fields who are interested in the future possibility of complex systems approach.

I am looking forward to meeting you in Beppu, Oita.

H. Tanaka

December 20, 2003

TIME TABLE

	RoomA	RoomB	RoomC	RoomD
1/27(Tue) 8:00				
13:00	Registration (Registration Desk)			
17:30	Welcome Party (in Hotel Arthur 10th Floor)			
1/28(Wed.) 8:00	Registration (Registration Desk)			
9:00	IPS (4) Chair: J. Wang	IS1 (5) Chair: T. Yamamoto	GS1(6) Chair: P. Sapaty	
10:30	Coffee Break			
10:40				Opening Ceremony
11:00				Plenary Talk Chair J. Johnson PT1 H. H. Lund
12:00	Lunch			
13:00	GS19 (6) Chair: J. Casti	GS13 (5) Chair: W. Wells	IS8 (6) Chair: M. Kono	
14:30	GS9 (4) Chair: M. Rizon	GS24 (4) Chair: J. Johnson	GS6 (4) Chair: H. H. Lund	
15:30	Coffee Break			
15:45	GS4 (6) Chair: C. Tayor	GS21 (7) Chair: D. J. G. James	IS7 (6) Chair: J. M. Lee	
17:30				

GS: General Session IS: Invited Session IPS: Invited Professor's Session PS Poster Session

GS1 Artificial Brain -I
GS2 Artificial Brain- II
GS3 Artificial Intelligence
GS4 Artificial Life-I
GS5 Artificial Life- II
GS6 Artificial Living
GS7 Bioinformatics
GS8 Cognitive Science
GS9 Computer Graphics
GS10 Evolutionary Computations-I
GS11 Evolutionary Computations - II
GS12 Genetic Algorithms
GS13 Human-Machine Cooperative System
GS14 Intelligent Control and Modeling-I
GS15 Intelligent Control and Modeling- II
GS16 Intelligent Control and Modeling- III
GS17 Micro-Robot World Cup-Soccer Tournament
GS18 Molecular Biology-I
GS19 Mobile Vehicles-I
GS20 Mobile Vehicles- II
GS21 Neural Networks
GS22 Robotics-I
GS23 Robotics- II
GS24 Robotics- III
IS1 Soft Robotics and Intelligence
IS2 Artificial and Natural Adaptions
IS3 Biological Information Processing and Control
IS4 Biomimetic Machines and Their Applications
IS5 Intelligent Pattern Classification
IS6 Intelligent Information Retrieval
IS7 Intelligent Mobile Robot
IS8 Computer Vision and Distributed Control Systems
IS9 Molecular Biology- II
IPS Invited Professor's Session
PS Poster Session

	RoomA	RoomB	RoomC	RoomD
1/29(Thu.) 8:00	Registration (Registration Desk)			
9:00	GS12(6) Chair: Y. G. Zhang	GS22 (5) Chair: C. Zhang	GS20 (6) Chair: J. M. Lee	
10:00				
10:30	Coffee Break			
11:00				Plenary Talk Chair D. J. G. James PT2 R. Pfeifer
12:00	Lunch			
13:00	PS (3)	IS3 (5) Chair: K. Abe	IS2 (5) Chair: K. Naitoh	
14:15	Coffee Break			
14:30	GS8 (6) Chair: K. B. Sim	GS14 (5) Chair: Z. Bubnicki	IS5 (4) Chair: S. Omatu	
15:45		GS5 (6) Chair: H. Kashiwagi	GS23 (7) Chair: T. Kubik	
16:00	GS10 (5) Chair: L. Han			
17:30				
18:00	AROB Award Ceremony (Chair: K. Watanabe) Banquet (Hotel Shiragiku)			
20:30				

GS: General Session IS: Invited Session IPS: Invited Professor's Session PS Poster Session

GS1 Artificial Brain-I
GS2 Artificial Brain- II
GS3 Artificial Intelligence
GS4 Artificial Life-I
GS5 Artificial Life- II
GS6 Artificial Living
GS7 Bioinformatics
GS8 Cognitive Science
GS9 Computer Graphics
GS10 Evolutionary Computations-I
GS11 Evolutionary Computations - II
GS12 Genetic Algorithms
GS13 Human-Machine Cooperative System
GS14 Intelligent Control and Modeling-I
GS15 Intelligent Control and Modeling- II
GS16 Intelligent Control and Modeling- III
GS17 Micro-Robot World Cup-Soccer Tournament
GS18 Molecular Biology-I
GS19 Mobile Vehicles-I
GS20 Mobile Vehicles- II
GS21 Neural Networks
GS22 Robotics-I
GS23 Robotics- II
GS24 Robotics- III
IS1 Soft Robotics and Intelligence
IS2 Artificial and Natural Adaptions
IS3 Biological Information Processing and Control
IS4 Biomimetic Machines and Their Applications
IS5 Intelligent Pattern Classification
IS6 Intelligent Information Retrieval
IS7 Intelligent Mobile Robot
IS8 Computer Vision and Distributed Control Systems
IS9 Molecular Biology- II
IPS Invited Professor's Session
PS Poster Session

	RoomA	RoomB	RoomC	RoomD
1/30(Fri.) 8:00	Registration (Registration Desk)			
9:00	GS15 (6) Chair: E. Hayashi	GS3 (5) Chair: T. Yamamoto	GS11 (4) Chair: H. Tanaka	
10:30	Coffee Break			
10:45	IS4 (5) Chair: K. Watanabe	GS2 (4) Chair: T. Arita	IS9 (4) Chair: K. Ohnishi	
12:00	Lunch			
13:00	GS18 (4) Chair: H. Hirayama	GS17 (3) Chair: M. Tanaka-Yamawaki	IS6 (4) Chair: H. Yanagimoto	
14:00	GS7 (4) Chair: K. Uosaki	GS16 (5) Chair: K. Watanabe		
15:15				
15:30	Farewell Party (Room A)			
16:30				

GS: General Session IS: Invited Session IPS: Invited Professor's Session PS Poster Session

GS1 Artificial Brain-I
GS2 Artificial Brain- II
GS3 Artificial Intelligence
GS4 Artificial Life-I
GS5 Artificial Life- II
GS6 Artificial Living
GS7 Bioinformatics
GS8 Cognitive Science
GS9 Computer Graphics
GS10 Evolutionary Computations-I
GS11 Evolutionary Computations - II
GS12 Genetic Algorithms
GS13 Human-Machine Cooperative System
GS14 Intelligent Control and Modeling-I
GS15 Intelligent Control and Modeling- II
GS16 Intelligent Control and Modeling- III
GS17 Micro-Robot World Cup-Soccer Tournament
GS18 Molecular Biology-I
GS19 Mobile Vehicles-I
GS20 Mobile Vehicles- II
GS21 Neural Networks
GS22 Robotics-I
GS23 Robotics- II
GS24 Robotics- III
IS1 Soft Robotics and Intelligence
IS2 Artificial and Natural Adaptions
IS3 Biological Information Processing and Control
IS4 Biomimetic Machines and Their Applications
IS5 Intelligent Pattern Classification
IS6 Intelligent Information Retrieval
IS7 Intelligent Mobile Robot
IS8 Computer Vision and Distributed Control Systems
IS9 Molecular Biology- II
IPS Invited Professor's Session
PS Poster Session

TECHNICAL PAPER INDEX

January 28 (Wednesday)

Room D

11:00~12:00 PT-1 Plenary Talk1
Chair J. Johnson

January 29 (Thursday)

11:00~12:00 PT-2 Plenary Talk2
Chair D. J. G. James

January 28 (Wednesday)

Room A

9:00~10:30 IPS Invited Professor's Session
Chair: J. Wang (Oita University, Japan)

13:00~14:30 GS19 Mobile Vehicles-I
Chair: J. Casti (Santa Fe Institute, USA)

14:30~15:30 GS9 Computer Graphics
Chair: M. Rizon (Northern Malaysia University College of Engineering, Malaysia)

15:45~17:30 GS4 Artificial Life-I
Chair: C. Tayor (University of California, Los Angels, USA)

Room B

9:00~10:30 IS1 Soft Robotics and Intelligence
Chair: T. Yamamoto (University of the Ryukyus, Japan)
Co-Chair: K. Yamada (University of the Ryukyus, Japan)

13:00~14:30 GS13 Human-Machine Cooperative System
Chair: W. Wells (University of Nevada, Las Vegas, USA)

14:30~15:30 GS24 Robotics-Ⅲ
Chair: J. Johnson (The Open University, UK)

15:45~17:30 GS21 Neural Networks
Chair: D. J. G. James (Coventry University, UK)

Room C

9:00~10:30 GS1 Artificial Brain-I
Chair: P. Sapaty (National Academy of Ukraine, Ukraine)

13:00~14:30 IS8 Computer Vision and Distributed Control System
Chair: M. Kono (University of Miyazaki, Japan)
Co-Chair: M. Yokomichi (University of Miyazaki, Japan)

14:30~15:30 GS6 Artificial Living
Chair: H. H. Lund (University of Southern Denmark, Denmark)

15:45~17:30 IS7 Intelligent Mobile Robot
Chair: J-M. Lee (Pusan National University, Korea)

January 29 (Thursday)

Room A

9:00~10:30 GS12 Genetic Algorithms
Chair: Y. G. Zhang (Academia Sinica, P.R.China)

13:00~14:15 PS Poster Session

14:30~16:00 GS8 Cognitive Science
Chair: K-B. Sim (Chung-Ang University, Korea)

16:00~17:30 GS10 Evolutionary Computations-I
Chair: L. Han (Beijing Technology and Business University, P.R.China)

Room B

9:00~10:30 GS22 Robotics-I
Chair: C. Zhang (Tsinghua University, P.R.China)

13:00~14:15 IS3 Biological Information Processing and Control
Chair: K. Abe (Tohoku University, Japan)
Co-Chair: N. Homma (Tohoku University, Japan)

14:30~15:45 GS14 Intelligent Control and Modeling-I
Chair: Z. Bubnicki (Wroclaw University of Technology, Poland)

15:45~17:30 GS5 Artificial Life-Ⅱ
Chair: H. Kashiwagi (Kumamoto University, Japan)

Room C

9:00~10:30 GS20 Mobile Vehicles-Ⅱ
Chair: J-M. Lee (Pusan National University, Korea)

13:00~14:15 IS2 Artificial and Natural Adaptions
Chair: K. Naitoh (Yamagata University, Japan)
Co-Chair: T. Ohira (Sony Computer Science Laboratories, Japan)

14:30~15:45 IS5 Intelligent Pattern Classification
Chair: S. Omatu (Osaka Prefecture University, Japan)

15:45~17:30 GS23 Robotics-II
Chair: T. Kubik (Wroclaw University of Technology, Poland)

January 30 (Friday)

Room A

9:00~10:30 GS15 Intelligent Control and Modeling-II
Chair: E. Hayashi (Kyushu Institute of Technology, Japan)

10:45~12:00 IS4 Biomimetic Machines and Their Applications
Chair: K. Watanabe (Saga University, Japan)
Co-Chair: K. Izumi (Saga University, Japan)

13:00~14:00 GS18 Molecular Biology-I
Chair: H. Hirayama (Asahikawa Medical College, Japan)

14:00~15:15 GS7 Bioinformatics
Chair: K. Uosaki (Osaka University, Japan)

Room B

9:00~10:30 GS3 Artificial Intelligence
Chair: T. Yamamoto (University of the Ryukyus, Japan)

10:45~12:00 GS2 Artificial Brain-Ⅱ
Chair: T. Arita (Nagoya University, Japan)

13:00~14:00 GS17 Micro-Robot World Cup-Soccer Tournament
Chair: M. Tanaka-Yamawaki (Tottori University, Japan)

14:00~15:15 GS16 Intelligent Control and Modeling-Ⅲ
Chair: K. Watanabe (Saga University, Japan)

Room C

9:00~10:30 GS11 Evolutionary Computations-II
Chair: H. Tanaka (Tokyo Medical & Dental University, Japan)

10:45~12:00 IS9 Molecular Biology-II
Chair: K. Ohnishi (Niigata University, Japan)

13:00~14:00 IS6 Intelligent Information Retrieval
Chair: H. Yanagimoto (Osaka Prefecture University, Japan)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Educational Robotics: Manipulative Technologies for Cognitive Rehabilitation

Henrik Hautop Lund*, Patrizia Marti°, Valentina Palma°

*Maersk Mc-Kinney Moller Institute for Production Technology
University of Southern Denmark, Campusvej 55, 5230 Odense M., Denmark
° Multimedia Communication Laboratory
University of Siena, Via dei Termini 6, 53100 Siena, Italy

hhl@mip.sdu.dk marti@unisi.it
www.adaptronics.dk

Abstract

The paper describes an early research related to the design, development and evaluation of a concept of flexible and physical components to manipulate conceptual structures. The concept exploits the affordances of physical objects in creating tangible media for collaborative and distributed edutainment spaces. Especially, we explore the possibility of allowing young children (and other potential users) to develop and interact with virtual/physical worlds by manipulating physical objects in different environments and at different scales (a classroom, a museum, a playground). A set of basic I-BLOCKS with individual processing and communication power was developed and can be further evolved in blocks of different shape and material in order to increase transparency of individual building block functionality. Users of the I-BLOCKS system can do 'programming by building' and thereby construct functionality of artefacts in an intuitive manner without the need to learn and use traditional programming languages. In the paper we describe in detail the technology of I-BLOCKS and discuss their application to a linguistic scenario.

Introduction

We developed a special kind of new devices for ambient intelligence solutions in edutainment environments [2]. We call these devices I-BLOCKS (Intelligent Blocks), and they are the supporting devices for exploring the design of a concept of flexible and physical components to manipulate conceptual structures. Here, we will describe a particular application of the I-BLOCKS technology to a linguistic scenario for children with dyslexia. In this scenario we transfer a well-known task used by speech therapists to the I-BLOCKS, in order to give more feedback and more sensorial information to the child.

The I-BLOCKS tool consists of a number of 'intelligent' building blocks that can be manipulated to create both physical functional and conceptual structures [2, 3]. The I-BLOCKS support our more philosophical claim that both body (physical structure) and brain (functional structure) play a crucial role in intelligence. The focus on building both physical and functional structures with the I-BLOCKS also lead to the possibility of investigating the concept of 'programming by building' [2], in which programming of a specific behaviour simply consists of building physical structures known to express that specific behaviour. So, we suggest moving away from programming the artefacts with traditional programming languages, and instead provide methods that allow people to 'program by building' without the need for any a priori knowledge about programming languages. Indeed, we even suggest to completely removing the traditional host computer (e.g. a PC) from the creative process.

The I-BLOCKS tool that support investigation of this innovative way of manipulating conceptual structures consists of a number of 'intelligent' building blocks (I-BLOCKS) that each contain processing and communication capabilities. Each I-BLOCK has a physical expression (e.g. a cube or a sphere). When attaching more I-BLOCKS together, a user may create a physical structure of I-BLOCKS that process and communicate with each other, depending on how the I-BLOCKS are physically connected to each other. Interaction with the surrounding environment happens through I-BLOCKS that obtain sensory

input or produces actuation output. So the overall behaviour of an 'intelligent artefact' created by the user with the I-BLOCKS depends on the physical shape of the creation, the processing in the I-BLOCKS, and the interaction between the creation and the surrounding environment (e.g. the user him/herself) – see [1] for details.

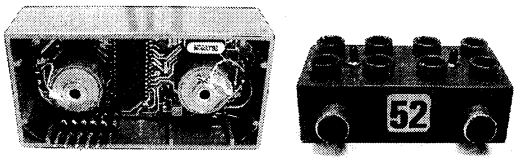


Fig. 1. Left: the internals of a building block with micro processor and communication channels. Right: example of *input (sensor) building block* that contains two microphones. © H. H. Lund, 2002.

Our first implementation of I-BLOCKS uses an electronic circuit containing a PIC16F876 40-pin 8 bit CMOS Flash microcontroller for processing, and provides four 2-ways serial connections in each I-BLOCK for communication (see Fig. 1). In order to better visualise the concept, we have chosen to make the housing out of rectangular LEGO DUPLO[1] bricks. Energy power from a battery building block is transported through the construction of I-BLOCKS via connectors in the corners on the bottom on each block and connectors in the studs on top of each block.

There exist different types of I-BLOCKS that all share the same standard technology of providing processing and communication capabilities. We term these *standard building blocks*. In a number of cases, the standard building blocks are extended with the addition of sensors in order to become *input building blocks*, and in a number of cases extended with the addition of actuation in order to become *output building blocks*. Input building blocks include building blocks with LDR sensors, IR sensors, microphones, switches, potentiometer, and output building blocks include building blocks with servo motor, DC motor, IR emitter, LEDs, sound generator, etc. (see examples on Fig. 1).

In order to verify the technological possibilities of the I-BLOCKS (e.g. versatility of control methods), we implemented different kinds of processing in the I-BLOCKS, making the I-BLOCKS becoming arithmetic blocks, behaviour blocks, neural blocks, and spiking neural blocks [5].

Several basic experiments tell us that I-BLOCKS may allow 'programming by building', and here we elaborate on how construction/configuration of I-BLOCKS can represent a conceptual structure, and be used at different scales and contexts.

I-BLOCKS for cognitive rehabilitation: an application scenario

Our new concept of flexible and physical components to manipulate conceptual structures allows exploration of scenarios at different scales and contexts of use. The above mentioned examples of I-BLOCKS implementation indicated use at object scale with small elements (LEGO DUPLO bricks) for individual use supporting gesture based interaction and direct manipulation. These small elements could, for instance, be used for arithmetic training by allowing the trainee to physically manipulate the structures of *arithmetic I-BLOCKS* and receive feedback directly from the structure. But they could also be used in unstructured activities, for instance where a child is simply performing free play by constructing with *behaviour I-BLOCKS* or *neural I-BLOCKS*, and interact with the construction that he/she builds.

In the following we present a particular application of the I-BLOCKS technology to a linguistic scenario. In this scenario we transfer a well-known task used by speech therapists to the I-BLOCKS, in order to give more feedback and more sensorial information to the child. Our hypothesis is to test whether external representations, in the form of dynamic I-BLOCK constructions, would assist children in learning linguistic structures in a more effective way than with the combination of static iconic pictures like the ones currently used by speech therapists. Besides this, we want to test if bricks, differently from cards, can support the children in the performance of logic and grammatical abstraction tasks. A third hypothesis is that bricks can support the creative activity of linguistic production once the structuring of sentence is well interiorized by children through the feedback provided by the I-BLOCKS.

Currently, the speech therapist tries to teach to children with language problem the right structure of a sentence. In focus group interviews, the speech therapist's work with children was

[1] LEGO and LEGO DUPLO are trademarks of LEGO System A/S.

observed, and it is evident that manipulation of objects is a very important feature in order to reach language skills. Every task has the form of a game, in which the speech therapist helps the child, giving good scaffolding to the task.

Children with dyslexia, i.e. a difficulty in the scholastic learning, or with a Specific Language Impairment (SLI), can have problems to understand the structure of a sentence, and the speech therapist tries to help using lots of instruments. One of a task that the speech therapist purveys is to construct a sentence with special cards. These cards have different shapes: (1) Small and tall green rectangle for article (Fig.2), (2) Rectangle with an icon on for noun (Fig.3), (3) Red arrow for verb (Fig.2), and (4) Small square for preposition.

Fig. 2: Example of an article and a verb card.

Fig.3: Example of a situation-card.

The activity is articulated in the following steps:

- The child has to recognize the different parts of a sentence without icons (verb and article).
- The therapist encourages the child to choose a situation depicted on the situation cards.
- The therapist presents the components of the specific sentence that the child has to identify and to use to structure the sentence (e.g. "the child eats the chicken").
- The child composes the sentence putting the constituting elements in sequence.
- the therapist asks the child to read the sentence putting his/her finger on the cards. If the performance is not correct, the therapist encourages the child to try again making questions like -Where is the word "the" in the following sequence? (Fig.4).

Another version of the task is to make available more noun-icons and to ask to the child to construct a good sentence. (semantic task).

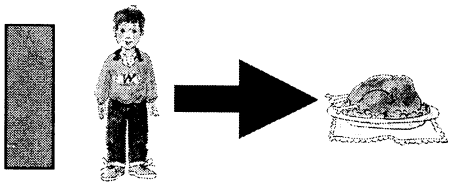


Fig. 4: An example of sentence where an article is missing ('The boy eats chicken').

In this task, the child manipulates directly the cards, and every card represents a well specified part of the sentence. The feedback, obviously, comes from the speech therapist. However, with the I-BLOCKS, we may develop a system where the child is manipulating the structure of sentences when manipulating with the physical structure of I-BLOCKS, and at the same time receives feedback from the I-BLOCKS construction. In the first mock-ups implementation, we have decided to preserve the characteristic of article (the small dimension), and to give different colours for different roles in the sentence: (1) Red small brick for article, (2) Green brick for noun, (3) Yellow display brick for verb.

So a sentence construction could take the form shown in figure 5. The I-BLOCKS can now give feedback and sensorial information to the child based upon the physical structure that the child has created.

The first testing activity was performed with two children, T. 6 years old, with hypoacusia and M. 10 years old, with dyslexia. In order to experiment the idea at a low cost, the Wizard of Oz technique was used. This technique involves a user interacting with a fake system (low fi prototypes) which is actually operated by a hidden developer - referred to as the 'wizard' [4]. The wizard processes input from a user and simulates system output. During this process the user is led to believe that they are interacting directly with the system. This form of prototyping is beneficial early on in the design cycle and provides a means of studying a user's expectations and requirements. The approach is particularly suited to exploring design possibilities in systems which are demanding to implement such as those that feature intelligent interfaces incorporating agents, advisors and/or natural language processing. In our case, we re-created a situation in which child had the idea of

interacting with a fully working I-BLOCK system that provided timely and appropriate feedback.

For the testing different verb, article and noun bricks were developed for composing Subject – Verb – Object sentences.

The situation cards represented the following situations:

- the child drinks the juice
- the child eats the chicken
- the child greets a friend
- the iceman sells the ice-ream
- the child buys the ice-cream
- the newsagent sells the newspapers
- the child looks at the television (in italian the sentence is without preposition! Il bambino guarda la televisione)
- the doctor examines the child
- the mum kisses the child
- the teacher reads a novel
- the mechanic repairs the car
- the grandma sews the dress
- the grandpa buys the bread
- the daddy drives the car
- the mum irons the trousers

In addition to noun, verb and article bricks, another special silver brick was used to verify the correct composition of the sentence. The silver brick produced a sound in case the sentence was incorrect, or it read the sentence without inflections in case the task was performed correctly.

The testing activity was similar to the one executed through the cards. The child composed the sentence using the bricks and then the therapist asked the child to read the sentence and check the correctness with the silver brick.

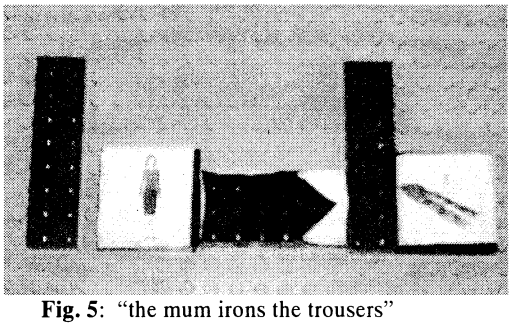


Fig. 5: "the mum irons the trousers"

In case of error, the child was encouraged to try other combinations until the correct one is reached. Therefore "system" pronounced the correct sentence so the child could compare is performance with the one of the system.

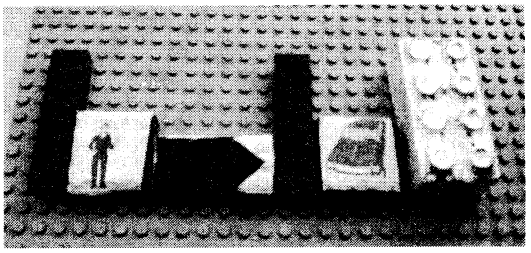


Fig. 6: a sentence with the silver brick

The Wizard of Oz worked well, so the mock-up was implemented for further testing. The experience revealed a number of interesting properties of the bricks:

- The interactive bricks sustained trial-and-error activity. The children were stimulated to seek different configurations of bricks and rapidly check the results.
- Children were in control of the experimental setting. They could check themselves the results of the activity without the support of the therapist.
- Children were much more involved in the activity. The same task performed with cards resulted boring and in some cases frustrating.
- The silver brick was always put at the end of the sentence to mean that the children understood the sequential structure of the sentence.
- The sound feedback is informative and not intrusive. It was interesting to note how the children were encouraged to reflect on the structure of the sentence in case of error.
- The abstraction task was executed correctly. The children became familiar with the bricks and were able to recognise the grammatical elements of the sentence (noun brick, article brick etc.).

In a later testing session the children were asked to invent a sentence using the bricks, without icons. The result was amazing. M. for example composed the following sentences that reflected his interest for animals:

The varan lives in Komodo
The varan eats goats in Komodo

where he used yellow bricks to represent the preposition (Fig. 7).

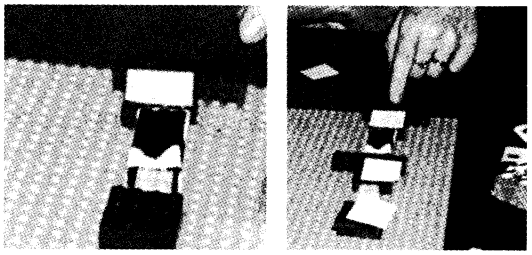
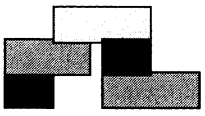


Fig. 7: creation of new sentences

Based on this positive feedback from the 'Wizard of Oz' experiments, we implemented a system in the I-BLOCKS reflecting the properties of the Wizard of Oz experiments. I-BLOCKS implemented in LEGO Duplo bricks would represent nouns, verbs, articles, preposition, based on their colour or size. Each block would also be labelled with the word. Further, a large I-BLOCK (in fact two blocks put together) would represent the magic silver brick. This large block would write the sentence on a display and play a happy or sad tune, depending on the correctness of the constructed sentence.
This implementation was tested with children from Denmark, Italy, and Tanzania. The initial experience suggests us that such an implementation may be useful with users with no learning problems (experiments performed at Aalokke Skolen, Odense, Denmark, and Tumaini University, Iringa, Tanzania), whereas the building may pose some problems to children with dyslexia (experiments performed in Siena, Italy in the context described above). We will investigate this difference further in the future.

Discussion

Following the Distributed Cognition approach [1], our hypothesis in designing I-BLOCKS for the linguistic scenario was that it may be possible to enhance cognition by mapping problem elements (components of a sentence) to an external, manipulative, physical and reacting construction in such a way that solutions become immediately evident and the children can receive feedback on their action of combining I-BLOCKS. Zhang and Norman [6] propose a theoretical framework in which internal representations and external representations form a "distributed representational space" that represents the abstract structures and properties of the task in "abstract task space" (p. 90). They developed this framework to support rigorous and formal analysis of distributed cognitive tasks and to assist their investigations of "representational effects [in which] different isomorphic representations of a common formal structure can cause dramatically different cognitive behaviours" (p. 88). "External representation are defined as the knowledge of the structure in the environment, as physical symbols, objects, or dimensions (e.g., written symbols, beads of abacuses, dimensions of a graph, etc.), and as external rules, constraints, or relations embedded in physical configurations (e.g., spatial relations of written digits, visual and spatial layout of diagrams, physical constraints in abacuses, etc.)" (p. 180) [7].



The/a cat eats the/a mouse
Il/un gatto mangia il/un topo
En kat spiser en mus

Fig. 8: Possible construction with the I-BLOCKS mock-up for the linguistic task in three different languages (English, Italian, and Danish).

The feature that different representations can cause dramatically different cognitive behaviour is referred to as "representational determinism" [7]. It is obviously easier to play tictactoe than 15. This is exactly because tictactoe relies on external representations utilizing what we are good at, whereas you need to represent the numbers and their sum internally in the other game which we do inefficiently.

The development of the setting for the linguistic scenario is still underway. Experiments with children with dyslexia will be performed to verify if the "representational determinism" of I-BLOCKS can effectively sustain linguistic learning.

Acknowledgement

Part of the implementation work was performed by J. Nielsen, S. Jensen, M. Pedersen. A. Rullo collaborated on the user research and testing with children. They all provided valuable discussions and contributed to the definition of the project vision. The work is partly sponsored by the Danish National Research Council project 'Intelligent Artefacts'.

References

[1] Hutchins, E. L. Cognition in the wild, MIT Press, London, 1995.
[2] Lund, H. H. Intelligent Artefacts. In Sugisaka and Tanaka, editors, Proceedings of 8th International Symposium on Artificial Life and Robotics. ISAROB, Oita, 2003.
[3] Lund, H. H. Neural Building Blocks. In 1st International IEEE EMB Conference on Neural Engineering, Capri, Italy, March 20-22 2003. IEEE Press.
[4] Maudsley, D., Greenberg, S. & Mander, R. Prototyping an intelligent agent through Wizard of Oz. INTERCHI '93 Conference Proceedings, pp. 277-284. 1993.
[5] Nielsen, J. and Lund, H. H. Spiking Neural Building Block Robot with Hebbian Learning. Proceedings of IROS'03. IEEE Press, 2003.
[6] Zhang, J., and Norman, D. A. Representations in Distributed Cognitive Tasks. Cognitive Science 18: 87-122, 1994.
[7] Zhang, J. The Nature of external Representations in Problem Solving. Cognitive Science 21:2, 179-217, 1997.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

I-Blocks in an African Context

Henrik Hautop Lund[1] Mikko Vesisenaho[2]

[1]Maersk Mc-Kinney Moller Institute for Production Technology
University of Southern Denmark, Campusvej 55, 5230 Odense M., Denmark

[2]Department of Computer Science,
University of Joensuu, P.O. Box 111, 80101 Joensuu, Finland

hhl@mip.sdu.dk mvaho@cs.joensuu.fi
www.adaptronics.dk

Abstract

The paper describes some initial experiences in involving users from Tanzania in the development of I-BLOCKS technology for the African cultural context. Users of the intelligent building blocks (I-BLOCKS) can do 'programming by building' and thereby construct functionality of artefacts in an intuitive manner without the need to learn and use traditional programming languages. We allow both secondary school pupils and university students in Tanzania to investigate possibilities by building with I-BLOCKS within three scenarios, for (1) mathematical training, (2) emotion construction, and (3) language grammar training. Based on the experience, the pupils and students engaged in description of future development of the technology, e.g. by developing concept maps for African I-Blocks.

Introduction

Most IT tools are developed on a Western cultural background, which influences the direction in which these tools are developed – and rightfully so, since the tools are also mainly delivered to users with this cultural background. However, the tools may not necessarily have the same success when used in different cultural settings. The reason for this being that the tools may be context dependent, and users may have a different cultural background, different kind of logical thinking, different daily living context, etc. [2]. Here, we present a technological tool that may facilitate development in different cultural contexts, and we try to verify the suitability of the tool by collaboration with secondary school pupils and university students in Tanzania.

We developed a set of technological building blocks, I-Blocks, to support learning by construction, and more specifically to support "programming by building" [4]. By attaching a number of basic building blocks together, the user may construct an artefact that can both perceive input, process, and produce output. Here, it is important to emphasise that each building block is a basic building block, and the overall functionality is obtained by combining a number of building blocks into a whole system. By using building blocks that the user him/herself should combine together, we may indeed be able to apply the technology to different cultural contexts, since it is not the designer/engineer that produces the overall system as a black box, but it is the user him/herself that, based on his/her desires, construct the desired artefact by combining a number of building blocks.

I-Blocks

The intelligent Building Blocks, called I-Blocks, were developed to each contain a microprocessor [4]. In the applications described here, the housing of the I-Blocks takes the form of LEGO DUPLO bricks that each contains the electronics including the microprocessor. The microprocessor used in this case is the PIC16F876 40-pin 8 bit CMOS Flash microcontroller. Each I-Block has four communication channels, two on the bottom of each brick and two on top of each brick. So, when attached together, I-Blocks may be able to communicate with each other over the two-way serial communication channels.

We constructed a number of different I-Blocks. As the basis, we constructed a Basic I-Block that

contains microprocessor and communication channels, and all other I-Blocks are extensions of this basic I-Block. The extensions include Input I-Blocks and Output I-Blocks. Input I-Blocks may be Basic I-Blocks with IR sensors, microphones, touch sensors, LDR sensors, etc., whereas Output I-Blocks may be Basic I-Blocks with motor, sound generator, IR emitter, display, etc.

The functionality of a system of I-Blocks is defined by the way the system is physically constructed, what processing take place inside the individual I-Blocks, and the interaction with the surrounding environment. In the work presented here, the physical construction and the interaction is decided by the users in Tanzania, whereas the processing is initially pre-defined by us engineers in a number of applications. Based on the experience with these applications, the Tanzanian users developed concept maps and figures for the further development of other I-Blocks applications suitable for the Tanzanian context, i.e. African I-Blocks.

I-Block Applications

Initially we will look at three applications used in the first I-Block workshops in a secondary school and at a university in Tanzania. The three applications were I-Blocks for emotional constructions, I-Blocks for mathematical training, and I-Blocks for language grammar training. The first of these applications can be used for a free, explorative play, whereas the latter two were used in a more direct educational setting with a more direct subject aim (i.e. math or language).

I-Blocks for Emotional Construction

The I-Blocks for emotional construction were initially developed by J. Nielsen and Prof. H. H. Lund at the Maersk Institute, University of Southern Denmark in collaboration with A. Rullo and Prof.ssa P. Marti at the University of Siena [5]. The main idea behind the I-Blocks for emotional construction is the implementation of a system that allows the user to construct different emotional models by the different physical constructions of I-Blocks systems. So the emotional model will be defined by the physical aspects of the construction made by the user. In this way, the user can explore different emotional models and emotional expressions by experimentation with the construction of different physical aspects. To our best knowledge, this is quite different from most other robotic experiments with emotional models, since in all other cases, the models are pre-defined by the researcher/engineer, and the user is interacting with the pre-defined model. In these applications, the emotional expression of the robotic entity will change dependent on the sensory input provided to the system by the user, but the user is not constructing the system or the emotional model him/herself. However, with the I-Blocks for emotional construction, it becomes possible for the user to actually construct the system and the emotional model by simple, physical manipulation with the I-Blocks.

I-Blocks for Mathematical Exercises

The arithmetic blocks include blocks for addition, subtraction, multiplication and division, sensor input blocks for setting input values, and output blocks to present output values. As an example let us look at the task of calculation of the results of arithmetic expressions, and here specifically the task to present a result for the following expression:

$$(x + y) * z$$

where x, y and z are either standard sensory inputs or user set inputs (the user can set input values with sensor building blocks by pressing a switch or turning a potentiometer). The task has been solved correctly for a solution that presents the right result with regards to every possible input.

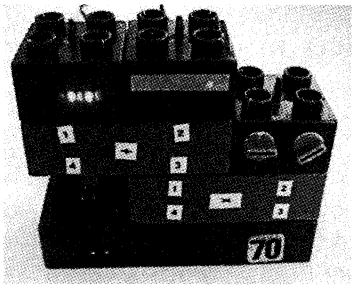


Fig. 1: A construction with arithmetic I-BLOCKS that performs an arithmetic calculation: (x+y)*z. The user can set the value of the small blocks on the top (x and y) and on the right side (z). The result is displayed as binary number with LEDs on the lower left block. © H. H. Lund, 2002.

This system with arithmetic blocks solves this task quite easily as shown in Fig. 1, because this system is built for arithmetic operation. The correct result will only be presented on the display (the block on the lower left corner) for as

long as the sub-results and the final result do not exceed the a value of 255, which is the maximum value possible for the value data type, and for communication in general.

The built structure can be seen in Fig. 1, where the binary display block on the lower left corner shows the result of adding two user set values (on the top) and then multiplying it with the user set value (on the right side on top of the multiplication block).

I-Blocks for Language Grammar Training

In the linguistic scenario we transfer a well-known task used by speech therapists to the I-BLOCKS, in order to give more feedback and more sensorial information to child [5]. We implemented a system in the I-BLOCKS reflecting the properties of Wizard of Oz experiments performed by Palma from Marti's group at University of Siena with children with dyslexia. I-BLOCKS implemented in LEGO Duplo bricks would represent nouns, verbs, articles, preposition, based on their colour or size. Each block would also be labelled with the word. Further, a large I-BLOCK (in fact two blocks put together) would represent a magic brick. This large block ('magic brick') would write the sentence on a display and play a happy or sad tune, depending on the correctness of the constructed sentence.

Tanzania and the Two Cooperation Institutions

Tanzania is an Eastern African country which has been independent since 1960s. There are about 130 different tribes and many tribal languages. The official languages are Swahili and English, which is the only teaching language since the beginning of the secondary. Although Tanzania is economically one of the weakest countries in the World, the demand and the will for development is wide. There is only 3,3 computers per 1000 people in Tanzania (2001), and demand for education and skilled teachers is large [7].

Tumaini University has been founded 1996. Iringa University Collage is the largest of its four campuses and is located in the southern highland of Tanzania. There are about 600 students studying law, business, journalism, theology and mathematics education in Iringa. 40 percent of students are female. The university is funded by Evangelical Lutheran Church in Tanzania.

Three years ago there were just quite old used computers from western countries. With the financial support of cooperational associations like ELCA (Evangelilic Lutheran Church in America) and FELM (Finnish Evangelic Lutheran Mission) the campus has got a new computer laboratory and local network with the Internet connection in addition to older computers. There are a couple courses related to computer applications.

Pommern Secondary is situated in a rural area, 60 kilometers from Iringa. The history of the village is very much related to the history of the mission station of the Pommern. The School is maintained by the Evangelic Lutheran Church of Tanzania, the Iringa Dioses. The amount of the students has been increasing all the time. There are 1200 students in Autunm 2003, and during the last years the amount of female students has been growing so fast that almost half of the new students are females. There is no telephone line and general electricity in Pommern.

Since the year 1998 there has been a computer laboratory in Pommern. The computer equipment of the Pommern is a gift of the Global Outreach association. There is a 10 laptop computers laboratory, which are using the electricity of the local solar power station. There is a computer teacher in Pommern, who has been trained for a semester in USA. Very rare secondary schools are able to serve computer education for students.

African Experiences with I-Blocks

Educational Aim of the Workshops

The educational methods in Tanzania are quite traditional. Teaching is mostly very authoritative, using "learning by heart"-method and it is not encouraging for problem solving. Students are not used to use computers, and programming by computers is not part of everyday life. The logic of programming is mostly related to western ways of Western system of knowledge, like inference, quantification, comparison, classification, representation, and measuring. Those might be hard to adopt [1].

Learning programming is quite hard for many persons in western countries, too (e.g. [3]). In this study we wanted to create a concrete, interactive problem solving process for learning programming in a cultural context. The I-Blocks we had for programming by building are still western products, and one of the main purposes was to evaluate the usefulness of them and to get feedback for future development of localized I-Blocks.

The workshops at Pommern Secondary and Tumaini University are a part of the development of culture contextual IT-education in partnership between Tumaini University Iringa University College, Tanzania and University of Joensuu, Finland [6].

Workshop at Pommern Secondary

Our 4-hour workshop at Pommern Secondary was 10th October, 2003. The workshop group was 11 students (ages 15-25) who have been using computers at least during some courses.

After very short overview of I-Blocks, students started to build quite freely using first arithmetical and then emotional I-Blocks. We, as researchers, were a bit scared how they would start with the "western" I-Blocks. Anyway they were very interested in building by I-Blocks. They built different constructions freely and got the straight feedback from the I-Blocks (Fig. 2,3).

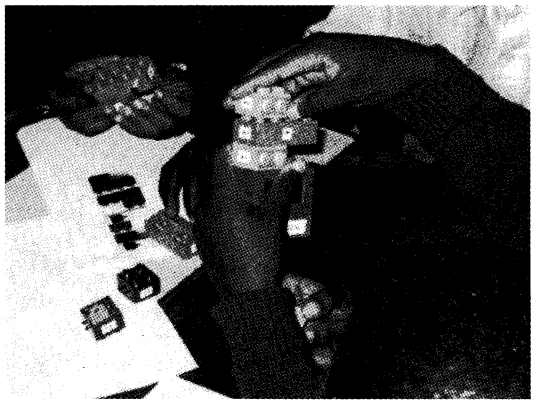


Fig. 2: Construction by I-Blocks at Pommern Secondary School, Tanzania.

Fig. 3: Pommern Secondary students programming by I-Blocks.

The language when using I-Blocks was not a problem, but when starting with the feedback questionnaires the English language was a certain problem, although English is the only teaching language in Tanzanian secondary schools. The local computer teacher translated the questions of questionnaire to local Swahili and encouraged then to use their imagination and draw pictures in addition to text when answering. Students filled the questionnaires very carefully. The whole session was video recorded.

Workshop at Tumaini University

At Tumaini University we had altogether seven hours workshop within three days, 13th-15th October, 2003. We had 27 second year journalism students (ages 21-42 years), who had computer experience of using office programs and web pages and email. The whole class was divided into four subgroups, and between different activities professor Lund gave a couple short lectures related to technology and robotics.

We started with arithmetical and emotional I-Blocks. We asked the students to build quite freely using the different I-Blocks. Students were working quite fast and they built differently working "towers" using the I-Blocks. The constructions they built were quite traditional ones, but the group work was very intensive and a couple of groups seemed to have very fun, too.

After using the arithmetical and emotional I-Blocks, they were using language blocks. Because they knew the English language quite well, it was a surprisingly easy task for them.

To get feedback, the students were asked to fill questionnaires in the end of the workshop. We encouraged the student to use freely their imagination when answering, and we reminded that there will not be a right or wrong answer. This is because at the Tanzanian school system, it is quite common just to have one exact right answer. In addition all the students drew a picture related to I-Blocks or the process of building I-Blocks and all the groups made and presented a concept map of African I-Blocks (Fig. 4). The working of the groups was video recorded.

Results

All the students stated that I-Blocks were useful. They thought they needed more education related to processes like building by I-Blocks, use of I-Blocks and generally related to IT-field.

One difference between the two groups was quite big: The secondary students seemed to have new inspiration and new ideas on how to use the I-Blocks. The university students needed more quickly additional exact exercises, like how you could construct a mathematical expression which outputs for example the result 22 or 45. This is probably somehow related to awareness to open problem solving.

The pictures related to text in the questionnaires showed that there are issues related to I-Blocks in Tanzania. Especially the secondary students thought that I-Blocks are very much related to building a house. That way, we notice that even LEGO DUPLO -based I-Blocks are somehow culture relevant for the Tanzanian culture.

Some university students' answers explained the relation between the structure of society and the I-Block construction process: Citizens are the batteries, parliament needs citizens' respect and ministers parliaments'. And the president is in the highest position. We can freely interpret that any of those can't work independently, but together those can do something which is meaningful and really working. It was speculated that the same is true with I-Blocks: one independent block is almost nothing, but constructed the right way they can work well together. This is also related to the basic logic of programming.

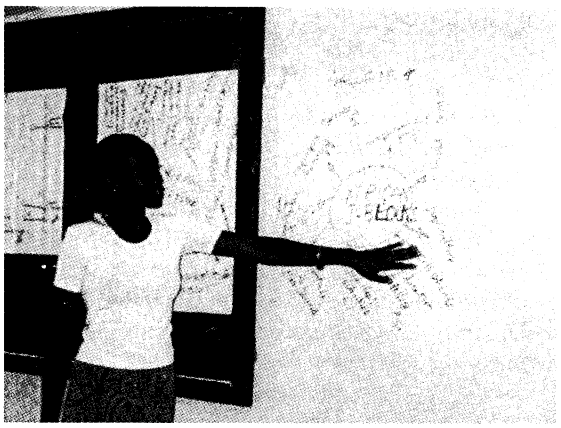


Fig. 4: Concept maps related to African I-Blocks' development.

The university students' concept maps - African I-Blocks - showed the very wide field related to robotics. The figures were quite general, but we can find the concepts like culture, entertainment, traffic lights, industries, health, agriculture, needs of schools, construction, cell phones, satellites etc. The concepts are quite general and mostly related to different needs of robotics for the society to develop.

Future Implications

The first tests of I-Blocks and programming by building in a developing country context showed that it is useful to add something concrete like the programming by building -workshop to be the first steps in learning programming. Next we will localize the I-Blocks, and add the possibility to program the microprocessors of I-Blocks. We will have the next workshops with the developed tools in Tanzania in March – April 2004.

Acknowledgements

Some of the scenarios were developed together with P. Marti's group at University of Siena. Part of the implementation work was performed by J. Nielsen, S. Jensen, M. Pedersen. Both Italian and Danish collaborators provided valuable discussions and contributed to the definition of the project vision. The work is partly sponsored by the Danish National Research Council project 'Intelligent Artefacts', and by the Academy of Finland project 'Information and Communication Technology Education for Development: a Tanzanian Perspective'.

References

[1] D'Ambrosio, U. Introduction: Ethno-mathematics and its First International Congress. *ZDM: Zentralblatt für Didaktik der Mathematik* 31 (2), 1999, 50-53.

[2] Duveskog, M., Sutinen, E., Tedre, M., and Vesisenaho, M. In search of contectual teaching of programming in a Tanzanian secondary school. In *Proceedings of Frontiers in Education Conference*, Boulder, Colorado, IEEE, 2003.

[3] Jenkins, T. Teaching Programming - A Journey from Teacher to Motivator. In *2nd Annual LTSN-ICS Conference, 2001.* http://www.ics.ltsn.ac.uk/pub/conf2001/papers/Jenkins%20paper.pdf (28.10.2003)

[4] Lund, H. H. Intelligent Artefacts. In Sugisaka and Tanaka, editors, *Proceedings of 8th International Symposium on Artificial Life and Robotics*. ISAROB, Oita, 2003.

[5] Marti, P., and Lund, H. H. Ambient Intelligence Solutions for Edutainment Environments. *In Proceedings of AI*IA 2003* (Eight National Congress of Associazione Italiana per l'Intelligenza Artificiale), Springer-Verlag, 2003.

[6] Sutinen, E., Vesisenaho, M., and Virnes, M. e-Based and Contact-Based Computing Studies for Tanzania: Action Research from Challengies via Changes to Chances. In *Proceedings of the 7th International Working Conference of International Federation and Information Processing (IFIP), Information Technology in Developing Countries (WG 9.4)*, IFIP, 2002, 444-452.

[7] *World Development Indicators*. Year 2001. World Bank Group. http://www.worldbank.org/data/countrydata/countrydata.html (18.11.2003)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Interacting with the real world – design principles for intelligent systems[1]

Rolf Pfeifer
Artificial Intelligence Laboratory, Department of Information Technology, University of Zurich
and
Department of Mechano-Informatics, School of Information Science and Technology, University of Tokyo

Abstract

The last two decades in the field of artificial intelligence have clearly shown that true intelligence always requires the interaction of an agent with a real physical and social environment. The concept of embodiment that has been introduced to designate the modern approach to designing intelligence has far-reaching implications. Rather than studying computation alone, we must consider the interplay between morphology, materials, brain (control), and the environment. A number of case studies are presented, and it is demonstrated how artificial evolution and morphogenesis can be used to systematically investigate this interplay. Taking these ideas into account requires entirely novel ways of thinking, and often leads to surprising results.

1. Introduction

In the traditional paradigm cognition, or generally intelligence, has been viewed as computation. The last two decades of research in the field have shown the limitations of this approach: true intelligence always requires the interaction with a real physical and social environment. An analysis of the failures of the traditional approach towards understanding and designing intelligent systems yields a fundamental neglect of the system-environment interaction. In contrast to a virtual or formal world (like chess, logic, or a virtual machine) the real world does not have precisely defined states, there is always only limited information available, there is only partial predictability, the environment has its own dynamics, and what an agent can do is not (completely) defined by the current situation. The interaction with the environment is always mediated by a physical body, with a particular morphology, i.e. body shape, and sensors and actuators distributed on the body. The concept of embodiment that has been introduced to designate the modern approach to designing intelligence has far-reaching implications. Rather than studying computation alone, we must consider the interplay between morphology, materials, brain (control), and the environment. These considerations go far beyond the trivial meaning of embodiment that "intelligence requires a body". They not only necessitate the interdisciplinary cooperation of computer science, neuroscience, engineering, and material science, but require entirely novel ways of thinking. In the context of this symposium on real world information systems, it is interesting to note that agents do not "get" the information from the environment, but they have to actively acquire it through specific kinds of interactions, so-called sensory-motor coordinations, as will be argued below.

As a first step towards a theory of intelligence based on the concepts of embodiment, a set of design principles for intelligent systems, has been proposed which can be grouped into two categories, design procedure principles, and agent design principles. The former are concerned with the way of proceeding, the latter with the actual design of agents. Examples of the former are "synthetic methodology", "time perspectives", "emergence", and "frame of reference", examples of the latter "cheap design", "ecological balance", and "sensory-motor coordination". Because of their relevance to real-world interaction the focus will be on "cheap design", "ecological balance", and "sensory-motor coordination"

We start by summarizing the principles. We then pick out three principles for illustration. We then briefly outline how to systematically explore the design principles using artificial evolution and morphogenesis. To conclude, a number of research challenges are proposed and some speculations are presented. It should be noted that this is not a technical paper but a conceptual one.

2. Design principles: Overview

There are different types of design principles: Some are concerned with the general "philosophy" of the approach. We call them "design procedure principles". Another set deals more with the actual design of the agent. The current overview given for reasons of completeness, will be very brief; a more extended version is in preparation (Pfeifer and Glatzeder, in preparation).

[1] *This paper which is prepared for AROB 9, 2004 is based on a lecture given for the COE Symposium on "Real World Information Systems", September 2004*

P-Princ 1: The synthetic methodology principle. The synthetic methodology, "understanding by building", implies on the one hand constructing a model – computer simulation or robot – of some phenomenon of interest (e.g. how an insect walks, or how we recognize a face in a crowd). On the other we want to abstract general principles (some examples are given below).

P-Princ 2: The principle of emergence. If we are interested in designing intelligent systems we should aim for emergence. Strictly speaking, behavior is always emergent, as it cannot be reduced to internal mechanism only; it is always the result of a system-environment interaction. In this sense, emergence is not all or none, but a matter of degree: the further removed from the actual behavior the designer commitments are made, the more we call the resulting behavior emergent.

P-Princ 3: The diversity-compliance principle. Intelligent agents are characterized by the fact that they are on the one hand exploiting the specifics of the ecological niche and on the other by behavioral diversity. In a conversation we have to comply with the rules of grammar, but then we can generate an infinite diversity of sentences.

P-Princ 4: The time perspectives principle. A comprehensive explanation of behavior of any system must incorporate at least three perspectives: (a) state-oriented, the "here and now", (b) learning and development, the ontogenetic view, and (c) evolutionary, the phylogenetic perspective.

P-Princ 5: The frame-of-reference principle. There are three aspects to distinguish whenever designing an agent: (a) the perspective, i.e. the one of the agent, the observer, or the designer; (b) behavior is not reducible to internal mechanism; trying to do that would constitute a category error; and (c) apparently complex behavior of an agent does not imply complexity of the underlying mechanism.

A-Princ 1: The three-constituents principle. Whenever designing an agent we have to consider three components. (a) the definition of the ecological niche (the environment), (b) the desired behaviors and tasks, and (c) the agent itself. The main point of this principle is that it would be a fundamental mistake to design the agent in isolation. This is particularly important because much leverage can be gained by exploiting the physical and social environment.

A-Princ 2: The complete agent principle. The agents of interest are autonomous, self-sufficient, embodied and situated. This view, although extremely powerful and obvious, is not very often considered explicitly.

A-Princ 3: The principle of parallel, loosely coupled processes. Intelligence is emergent from an agent-environment interaction based on a large number of parallel, loosely coupled processes that run asynchronously and are connected to the agent's sensory-motor apparatus.

A-Princ 4: The principle of sensory-motor coordination. Intelligent behavior (e.g. locomotion, perception, categorization) is to be conceived as a sensory-motor coordination. This sensory-motor coordination, in addition to enabling the agent to interact efficiently with the environment, serves the purpose of structuring its sensory input.

Table 1: Overview of the design principles

Number	Name	Description
		Design procedure principles
P-Princ. 1	Synthetic methodology	Understanding by building
P-Princ 2	Emergence	Systems designed for emergence are more adaptive
P-Princ 3	Diversity-compliance	Tradeoff between exploiting the givens and generating diversity solved in interesting ways
P-Princ 4	Time perspectives	Three perspectives required: "Here and now", ontogenetic, phylogenetic
P-Princ 5	Frame-of-reference	Three aspects must be distinguished: perspective, behavior vs. mechanisms, complexity
		Agent design principles
A-Princ 1	Three constituents	Task environment (ecological niche, tasks), and agent must always be taken into account
A-Princ 2	Complete agent	Embodied, autonomous, self-sufficient, situated agents are of interest
A-Princ 3	Parallel, loosely coupled procs	Parallel, asynchronous, partly autonomous procs, largely coupled through interaction with environment
A-Princ 4	Sensory-motor coordination	Behavior sensory-motor coordinated with respect to target; self-generated sensory stimulation
A-Princ 5	Cheap design	Exploitation of niche and interaction; parsimony
A-Princ 6	Redundancy	Partial overlap of functionality based on different physical processes

A-Princ 7	Ecological balance	Balance in complexity of sensory, motor, and neural systems: task distribution between morphology, materials, and control
A-Princ 8	Value	Driving forces; developmental mechanisms; self-organization

One of the powerful implications is that the problem of perception is greatly simplified through the interaction with the real world because the latter supports the generation of "good" patterns of sensory stimulation, "good" meaning correlated, and stationary (at least for a short period of time). One of the essential points here is that sensory stimulation is generated through the interaction with the environment – which is a physical process, not a computational one.

A-Princ 5: The principle of cheap design. Designs must be parsimonious, and exploit the physics and the constraints of the ecological niche. A trivial example is a robot with wheels which exploits the fact that the ground is mostly flat. Other examples are given below.

A-Princ 6: The redundancy principle. Agents should be designed such that there is an overlap of functionality of the different subsystems. Examples are sensory systems where, for example, the visual and the haptic systems both deliver spatial information, but they are based on different physical processes (electromagnetic waves vs. mechanical touch).

A-Princ 7: The principle of ecological balance. This principle consists of two parts: (1) Given a certain task environment, there has to be a match in the "complexity" of the sensory, motor, and neural system. (2) Given a particular task environment, there is a balance or task distribution between morphology, materials, and control (e.g. Hara and Pfeifer, 2000). Often, if the morphology and the materials are right, control will be much cheaper. Since we are dealing with embodied systems, there will be two dynamics, the physical one or body dynamics and the control or neural dynamics that need to be coupled. (e.g. Ishiguro et al., 2003).

A-Princ 8: The value principle. This principle is, in essence, about motivation. It is about why the agent does anything in the first place. Moreover, a value system tells the agent whether the result of an action was positive or negative (this is an very fundamental issue; there is no room for a comprehensive discussion here)..

Note that this set is of principles by no means complete. For example, a set of principles for designing evolutionary systems, is currently under development.

3. Illustrations of embodiment

The following examples are to illustrate that embodiment not only has physical implications, but important information theoretic ones. (concerning – neural – control)

The passive dynamic walker, the quadruped "Puppy", and the dancing robot "Stumpy"

The passive dynamic walker is a robot capable of walking down an incline without any actuation and control: it is "brainless", so to speak. In order to achieve this task the passive dynamics of the robot must be exploited. This kind of walking is very energy efficient and there is an intrinsic naturalness to it. However, its "ecological niche" (i.e. the environment in which the robot is capable of operating) is extremely narrow: it only consists of inclines of certain angles. To make this work, a lot of attention was devoted to morphology and materials. For example, the robot is equipped with wide feet of a particular shape to guide lateral motion, soft heels to reduce instability at heel strike, counter-swinging arms to negate yaw induced by leg swinging, and lateral-swinging arms to stabilize side-to-side lean (Collins et al., 2001).

The quadruped "puppy" by Fumiya Iida of the AILab of the University of Zurich, represents another example of exploitation of dynamic and of the interplay of morphology, materials, and control (fig. 1).

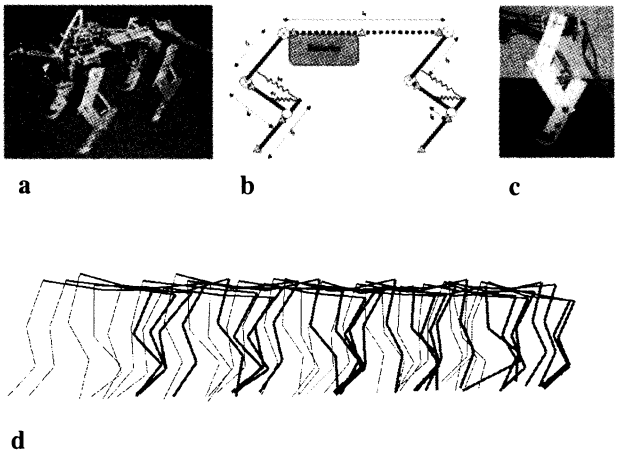
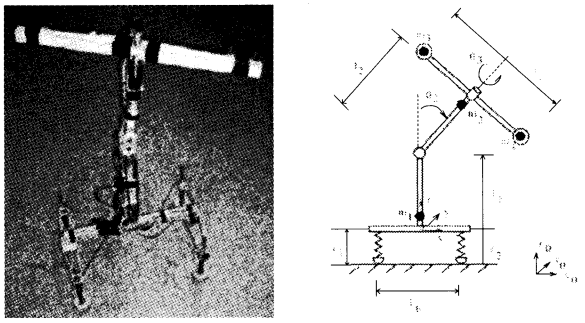


Figure 1. The quadruped "Puppy". (a) Picture of entire "puppy". (b) Diagram showing joints, servo-motor actuated joints [circles with crosses], and flexible spine [dotted line]. (c) The spring system in the hind legs. (d) Movement of hind legs [red] recorded at 25 frames per sec.

The legs perform a simple oscillation movement, but in the interaction with the environment, through the interplay of the spring system, the flexible spine (note that the battery is attached to the elastic spine which provides precisely the proper weight distribution), and gravity, a natural quadruped locomotion occurs, sometimes with all four legs up in the air. The system has self-stabilizing characteristics: there are on sensors on the robot.

For "Stumpy" (Paul et al., 2002) the goal was to generate a large behavioral diversity with as little control as possible. Stumpy's lower body is made of an inverted "T" mounted on wide springy feet (figure 2). The upper body is an upright "T" connected to the lower body by a rotary joint, the "waist" joint. The horizontal beam on the top is weighted on the ends to increase its moment of inertia. It is connected to the vertical beam by a second rotary joint, the "shoulder" joint. Stumpy's vertical axis is made of aluminum, while both its horizontal axes and feet are made of oak wood.

Stumpy can locomote in many interesting ways: forward in a straight or curved line or sideways, it has different gait patterns, and it can turn on the spot. Interestingly, this can all be achieved by actuating only two joints. The reason this works is because the dynamics, given by its morphology and its materials (elastic, spring-like materials, surface properties of the feet), is exploited in clever ways.



a b

Figure 2. The dancing, walking, and hopping robot Stumpy. (a) Photograph of the robot. (b) Schematic drawing (see text).

These three case studies illustrate the principles of cheap design and ecological balance. Loosely speaking, we can say that the control tasks, the neural processing, are partly (or completely, in the case of the passive dynamic walker) taken over by having the proper morphology and the right materials. Note that cheap design is not restricted to simple systems: it also applies to humans as highly complex biological creatures, as they also exploit the passive forward swing of the legs when walking.

Reaching and grasping – the principle of sensory-motor coordination as a key to higher levels of intelligence

Let us pursue this idea of exploiting the dynamics a little further. Human arms are built of muscles, tendons, ligaments, and bones, materials that are non-rigid to varying degrees. All these materials have their own intrinsic properties like mass, stiffness, elasticity, and damping, to mention but a few. These properties are all exploited in interesting ways in natural systems. For example, there is a natural position for a human arm which is determined by its anatomy and by these properties. Reaching for and grasping an object like a cup with the right hand is normally done with the palm facing left, but could also be done – with considerable additional effort – the other way around. Assume now that the palm of your right hand is facing right and you let go. Your arm will immediately turn back into its natural position. This is not achieved by neural control but by the properties of the muscle-tendon system (like a damped spring). Put differently, the morphology (the anatomy), and the materials provide physical constraints that make the control problem much easier – at least for the standard kinds of movements.

There is an additional point of central interest. Assume that you are grasping an object. Through the act of grasping, a lot of rich sensory stimulation is generated at the finger tips, and because of the anatomy, the grasped object – almost – automatically is brought into the range of the visual system. Grasping, like pointing and reaching, are processes of sensory-motor coordination. Sensory-motor coordination is subtended by anatomic (morphological) and material properties of the hand-arm-shoulder system, thus facilitating neural control. The sensory stimulation generated in this way implies correlations within and between sensory modalities, which is a prerequisite for developing higher levels of cognition. In this way, we are beginning to see how embodiment constitutes a precondition for intelligent behavior! The generation of structured sensory stimulation through physical interaction with the environment represents a key towards understanding developmental processes, as they are fundamental in humanoid robotics.

4. Artificial evolution and morpho-genenesis

We postulated and discussed a number of agent design principles. We also pointed out the principle of emergence. If we could demonstrate that the agent design principles would emerge from more fundamental evolutionary processes, this would corroborate the principles. As we are interested in embodied systems we must define processes capable of co-evolving morphology, materials, and control. This can be achieved through artificial evolution with morphogenesis based on genetic regulatory networks. This way, we can study agent design systematically and observe the – potential – emergence of agent designs. In order to provide a feel for the methodology, we are including a paragraph with a short description of how this works.

The mechanics of artificial genetic regulatory networks

We provide only a non-technical introduction for genetic regulatory networks, for details, see, e.g. Bongard and Pfeifer (2001), Bongard (2002, 2003). It should be stressed, that although this computational system is biologically inspired, it does not constitute a biological model; rather, it is a system in its own right. Also, when we use biological terminology, e.g. when we say that "concentrations of transcription factors regulate gene expression", this is meant metaphorically.

The basic idea is the following. A genetic algorithm is extended to include ontogenetic development by growing agents from genetic regulatory networks. In the example presented here, agents are tested for how far they can push

a large block (which is why they are called "block pushers"). Figure 3a shows the physically realistic virtual environment. The fitness determination is a two-stage process: the agent is first grown and then evaluated in its virtual environment. Figure 3b illustrates how an agent grows from a single cell into a multicellular organism.

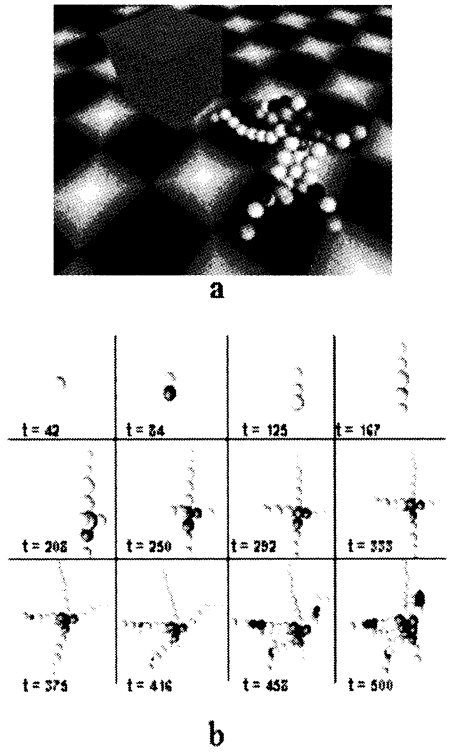


Figure 3. Examples of Bongard's "block pushers". (a) An evolved agent in its physically realistic virtual environment. (b) growth phase starting from a single cell, showing various intermediate stages (last agent after 500 time steps). Spheres with different shading indicate differentiation.

The algorithm starts with a string of randomly selected floating point numbers between 0 and 1. A scanning mechanism determines the location of the genes. Each gene consists of 6 floating point numbers which are the parameters that evolution can modify. They are explained in figure 4. There are transcription factors that only regulate the activity of other genes, there are transcription factors for morphology, and for neuronal growth. Whenever a gene is "expressed", it will diffuse a transcription factor into the cell from a certain diffusion site. The activity of this genetic regulatory network leads to particular concentrations of the transcription factors to which the cell is sensitive: whenever a concentration threshold is exceeded, an action is taken. For example, the cell may increase or decrease in size, if it gets too large, it will split, the joint angles can be varied, neurons can be inserted, connections added or deleted, structures can be duplicated, etc. The growth process begins with a single unit into which "transcription factors" are injected (which determines the primary body axis). Then it is left to the dynamics of the genetic regulatory network. The resulting phenotype is subsequently tested in the virtual environment. Over time, agents evolve that are good at pushing the block.

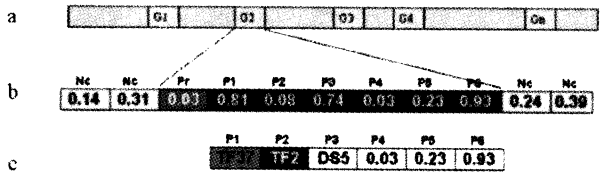


Figure 4. The mechanisms underlying the genetic regulatory networks. (a) Genes on the genome. Which regions are considered to be genes is determined by an initial scanning mechanism (values below 0.1 are taken as starting positions). (b) and (c) An example of a particular gene. Nc means "non-coding" region, Pr is a promoter site (start of gene), P1 through P6 are the parameters of the gene. P1: the transcription factor (TF) that regulates the expression of this gene (there are 20 different regulatory TSs, so this value is rounded into the range [0,19]. P2: the TF the gene emits if expressed (there are 43 TFs [20 regulatory and 23 that cause phenotypic change], so this value is rounded into the range [0,42]). P3: the diffusion site, i.e. the location in the cell from which the TF is diffused. P4: the quantity of TF emitted by this gene, if expressed. P5, P6: lower and upper bounds of the regulatory TF concentrations for which the gene is expressed.

4. Conclusions: Research challenges

Let us conclude by listing a few research challenges: (1) Theoretical understanding of (intelligent) behavior. In spite of half a century of research in artificial intelligence, we are still lacking a profound understanding of the mechanisms of intelligent behavior. With the set of design principles provided earlier, we hope to make a –however small – pertinent contribution. At the moment, these principles are qualitative in nature and a more quantitative formulation will be required in the future. (2) Achieving higher levels of intelligence through development. We only briefly touched upon sensory-motor coordination as a principle that is instrumental in achieving higher levels of intelligence. The field of "developmental robotics" capitalizes on this issue and we can expect many exciting results from it. However, the field in its current state is lacking a firm theoretical foundation. (3) Fully automated design methods (artificial evolution and morphogenesis). One of the big challenges is the automation of design. Designing embodied systems presents an additional challenge, as we need to take into account the interplay between environment (physical, but also social), morphology, materials, and control. (4) Moving into the real world (evolution, growth, etc.). To date, growth processes can only be achieved in simulation experiments – real world growth processes are only in their very initial stages in research laboratories and cannot yet be exploited for growing sophisticated creatures. This point represents an enormous challenge and will require many years of basic research.

Acknowledgments

I would like to thank Yasuo Kuniyoshi who initiated my visiting professorship at the 21st Century COE at the School of Information Science and Technology of the University of Tokyo and Masanori Sugisaka for inviting me to the AROB 9 conference.

References

Bongard, J.C. (2003). *Incremental approaches to the combined evolution of a robot's body and brain.* Unpublished PhD thesis. Faculty of Mathematics and Science, University of Zurich.

Bongard, J.C. (2002). Evolving modular genetic regulatory networks. In *Proc. IEEE 2002 Congress on Evolutionary Computation (CEC2002).* MIT Press, 305-311.

Bongard, J.C., and Pfeifer, R. (2001). Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. In L. Spector et al. (eds.). *Proc. of the Sixth European Conference on Artificial Life*, 401-412.

Collins, S.H., Wisse, M., and Ruina, A. (2001). A three-dimensional passive-dynamic walking robot with two legs and knees. *Int. J. of Robotics Res.*, 20, 607-615.

Ishiguro, A., et al. (2003). Toward a "well-balanced" design: a robotic case study. How should control and body dynamics be coupled? *Proc. AMAM-2003.*

Hara, F., and Pfeifer, R. (2000). On the relation among morphology, materials and control in morpho-functional machines. In *Procs. 6th SAB Conf.* 33-40.

Paul, C., Dravid, R. and F. Iida (2002b) Design and control of a pendulum driven hopping robot. *Proc IROS-2002,.*

Pfeifer, R., and Glatzeder, B. (in preparation). *How the body shapes the way we think: the embodied revolution in artificial intelligence.* Cambridge, Mass.: MIT Press.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

WHY THE FUTURE HAPPENS

Socionomics and the Science of Surprise

John L. Casti

Abstract

This talk presents the implications of recent work on wave-like patterns in social phenomena for characterizing and predicting the flow of human events and actions, such as the outcome of political elections, trends in films and fashion, the outbreak of war, and the rise and fall of civilizations. The talk will show that all such collective human social events are generated by the changing social mood in a population, and that the changes in this mood follow patterns that are predictable. This fact has led to the emerging field of socionomics, which is nothing less than a "science of surprise". The story of the development of socionomics and its mode of forecasting social trends will be presented through numerous examples and stories, as well as illustrations of how socionomics provides a systematic, coherent tool for predicting changing trends in the overall social mood. The talk shows that these trends exist on all time-scales—minutes to decades—and can be measured by the gyrations of financial market indexes, such as the Dow Jones Industrial Average. Thus, the talk illustrates how to actually predict "surprises", the turning points in social trends. So in a very real sense socionomics provides what amounts to a telescope for seeing the shape of the future.

The "take-home" message from this talk is twofold: The future is predictable in exactly the same probabilistic way that the weather is predictable, and that thoughts cause action and events—not vice-versa! Individual human thoughts are gathered together through the herding instinct hard-wired into every mammalian brain. These individual moods are like "bets" people place about the future. The bets then self-organize into an overall collective social mood, which after an appropriate period of time depending on the nature of the event, gives rise to things like wars, election results and styles in popular culture. This line of argument is exactly the opposite of that usually put forth by academic thinkers, Op-Ed writers, intellectual commentators, and other assorted pundits in their attempts to "explain" the flow of human events. The conventional explanations mostly go from "actions ? moods ? thoughts" instead proceeding in the "socionomic direction", which puts things exactly the other way around.

Introduction

Socionomics is about perceptions, the perceptions and beliefs that stand behind the explanations offered by the media, op-ed writers, intellectuals, and other assorted pundits for why collective human social events happen the way they do. In other words, explanations for why do we see what we do and not see something else. The taken-for-granted beliefs forming the cornerstone for these explanations give rise to what is customarily termed the "conventional wisdom." And it is not just megawattage events like political revolutions or world wars that the conventional analyst tries to explain, but also every type of trend in social behavior ranging from the style of lyrics in popular songs to the reelection of presidents to the rise and fall of the popularity of different types of social icons like sports heros and television personalities. The thread running through socionomics is the declarative fact that *the conventional wisdom is* **always** *wrong!* Not just "occasionally" or "sometimes" or "usually" wrong. But *unfailingly* wrong.

This is a strong claim and calls for the strongest possible arguments and examples to support it. Thus, socionomics is a radical theory, revolutionary even. Radical in that the it's core assertion is that the cause-and-effect structure of how events happen in the social and behavioral realm is exactly the opposite of how they unfold in the worlds of physics and the other natural sciences; revolutionary because this reversal of assumed realities that has become engrained in our thinking for centuries forces a revolution in the assumptions we make about the "scheme of things" by which we account for every type of collective social action we experience in daily life. To state it compactly, socionomics describes a revolution in the way we create the explanation of historical events as they unfold on all time scales.

Socionomics in Action

It's important to realize what socionomics offers is a predictive theory of social *trends;* it is not a theory of specific events. This is much in the same way that life insurance companies can predict very accurately how many people in different demographic categories will die each year, but cannot say which specific people will die.

Understanding socionomics requires comprehending the difference between the following two contrasting hypotheses:

A. **The Conventional Hypothesis:** Social mood is buffeted by economic, political and cultural trends and events. News of such events affects the social mood, which in turn affects people's readiness for investing.

B. **The Socionomic Hypothesis:** Social mood is a natural product of human interaction and is patterned according to something called the Wave Principle (explained more fully below). Its trends and extent determine the character of social action, including the economic, political and cultural trends and events.

The contrast between these two positions comes down to this: The conventional analyst argues that in the social setting, events govern mood; the socionomic analyst maintains that mood governs events. In both cases, the stock market is seen as an efficient mechanism. In the first instance, it presumably revalues stocks continually and rationally in reaction to events; in the second, it revalues stocks continually and impulsively as the independent social mood changes. stock market plays a central role in the theory of socionomics as it serves as a vehicle to measure social mood. I'd like to consider here just one example of a social event taken from everyday life, and see which of the two Hypotheses, A or B, is more strongly supported by the actual facts at hand.

In the social realm, socionomics provides a theory of social mood and trends, basically a theory of the *necessary;* chaotic dynamical systems determine what specific events or people will appear to manifest the overall social trend. With this as prelude, let me move on to the example.

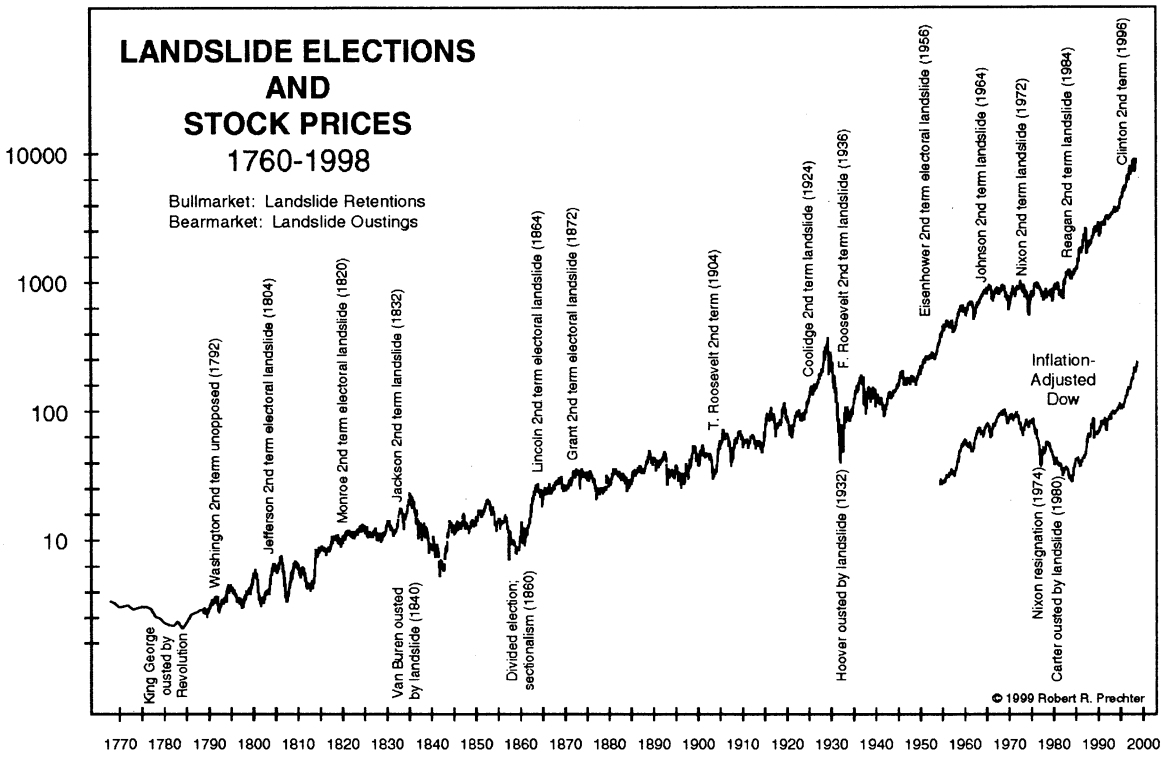
– Political Reelection –

The conventional wisdom is that political trends are a key determinant of the stock market's gyrations. As an election approaches, commentators endlessly debate the effect the outcome of the election will have on stock prices. Investors argue over which candidate is more likely to influence the market to go up or down. We often hear statements like, "If so-and-so gets elected, it will be good/bad for the market." If this causal relationship were valid, then there would be evidence that a transfer of power from one party's leader to another affects the stock market. There would also be evidence that certain political parties or policies reliably produce bull or bear markets. There is no study that shows any such connections or correlations.

A socionomist, on the other hand, can show the opposite flow of causality at work. Examining the figure above of presidential elections, it's easy to see that strong and persistent trends in the stock market determine (i.e., predict without fail) whether an incumbent president will be re-elected in a landslide or defeated in one. In all cases where an incumbent remained in office in a landslide, the stock market's trend was up. In all cases where an incumbent was rejected by a landslide, the stock market's trend was down. There is not a single case in which an incumbent won re-election despite a deeply falling stock market or lost in a landslide despite a strongly rising market.

The socionomic conclusion is clear: When social mood waxes positive, as reflected by persistently rising stock prices, citizens vote to retain the leader who symbolizes their upbeat feelings and who they presume helped cause the positive social mood. When the social mood turns negative, as reflected by persistently falling stock prices, voters decide to throw out the incumbent who symbolizes their downbeat feelings and who they presume helped cause the conditions leading to their pessimistic outlook. The political policies of the incumbent and his challenger are irrelevant to this dynamic. The key is a desire for change per se, not for any particular type of change. The conventional wisdom has no explanation for reconciling the relationship between these phenomena.

To conclude this example, socionomics asserts that an increasingly negative social mood impels voters to magnify politicians' weaknesses, minimize their strengths, perceive



that old political styles have failed, and eventually oust the incumbent at the next election or even before it occurs. A bear market has raged for three years. Voter hostility had to erupt somewhere.

Summarizing the overall lesson:

The **conventional** analyst asks: How do political leaders' actions cause a change in their popularity (social mood)?

The **socionomics** analyst asks: How does the social mood affect voters perceptions of political leaders?

The socionomic line of argument is certainly cause to ponder the fate of President George W. Bush in the forthcoming 2004 presidential election. If the social mood, i.e., financial markets, take off between now and then, Mr. Bush will be reelected; if they fall a long way by the autumn of 2004, then his political career and presidency is going to go the way of Van Buren, Nixon, Hoover, Carter and all the other political "dodo birds" of the species *politicus americanus.*

Summary

Socionomics persuasively argues that it is the collective social mood in a population that drives and generates collective human behavior and observable events—what we call "history"—and it has always been that way. To put it compactly, this book is devoted to the argument that social mood is the "engine" that creates history. To make this case, the book will draw upon psychological and neurobiological theories of herding behavior, ideas from complexity science as to how moods "emerge" in a population, and the way these moods cluster in networks, as well as the consequent translation of social mood and its observed wave patterns on all time scales into historical and economic events, not to mention shifts in popular culture. This constellation of ideas and examples explicates and validate the following diagram, which I term *The Fundamental Principle of Socionomics:*

herding impulse $\longrightarrow$ social mood $\longrightarrow$ collective human social events

Each component of the Fundamental Principle, including the arrows, will be dissected in detail, mostly by a plethora of stories and examples taken from every aspect of human social life. These accounts will be of the type presented above in abbreviated form, in order to show the reader how the various pieces of the Fundamental Principle fit together to generate events ranging from current popular films to the outbreak of warfare. Socionomics studies the way herding behavior arises as an evolutionarily advantageous way of surviving in a world of imperfect information. So if your interest is in when and why pop stars are made, when certain movie genres will be popular, when corporate scandals will hit the front pages of the paper, why Martha Stewart is being persecuted, why we went to war in Iraq, and how the next presidential election will turn out, then socionomics is the theory that will provide answers to these types of questions. It shows the way in which the social moods serve as previews of "coming attractions" and actions from actors as disparate as the Enron Corporation, Michael Jordan, and Osama bin Laden.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Micro-/Nanorobotic System

Toshio FUKUDA, Fumihito ARAI, and Lixin DONG

Department of Micro System Engineering, Nagoya University
Furo-cho 1, Chikusa-ku, Nagoya 464-8603, JAPAN
fukuda@mein.nagoya-u.ac.jp, arai@mein.nagoya-u.ac.jp, and dong@mein.nagoya-u.ac.jp

Abstract

Strategies leading from microrobotic system to nanorobotic system are proposed. A nano laboratory—a prototype nano manufacturing system—is presented, which is composed with a nanorobotic manipulation system with 4 units and 16 degrees-of-freedom (DOFs), a nano fabrication system based on electron-beam-induced deposition (EBID) with an internal or external precursor evaporation reservoir equipped with a thermal field emission electron source or a nanotube cold cathode, and a real-time observation and measurement system based on a field emission scanning electron microscope (FESEM) equipped with 3-4 conventional atomic force microscope (AFM) cantilevers, piezoresistive levers or nanotube probes. Nanotube devices including a mass flow sensor, a linear bearing, and nanotube scissors are fabricated in the nano laboratory.
Key words: carbon nanotube, nanodevice, microrobotics, nanorobotic manipulation, nano laboratory, electron-beam-induced deposition

1. Introduction

Two strategies towards the realization of nanotechnology have been presented, i.e., top-down and bottom up. The former one is mainly based on nanofabrication and includes technologies such as nano-lithography, nano-imprint, and etching. Presently, they are still 2D fabrication processes with low resolution. The later one is an assembly-based technique. At present, it includes such items as self-assembly, dip-pen lithography, and directed self-assembly. These techniques can generate regular nano patterns in large scales. To fabricate 3D complex nano devices there are still no effective ways by so far. Here we show our effort on the development of a nano laboratory, a prototype nanomanufacturing system, based on nanorobotic manipulations. In which, we take a hybrid strategy as shown in Fig.1. In this system, nano fabrication and nano assembly can be performed in an arbitrary order to construct nano building blocks and finally nano devices. The most important feature in this system is that the products can be fed back into the system to shrink the system part by part leading to nanorobots. Property characterization can be performed in each intermediate process. Due to the nanorobotic manipulation system, dynamic measurement can be performed rather than conventional static observations.

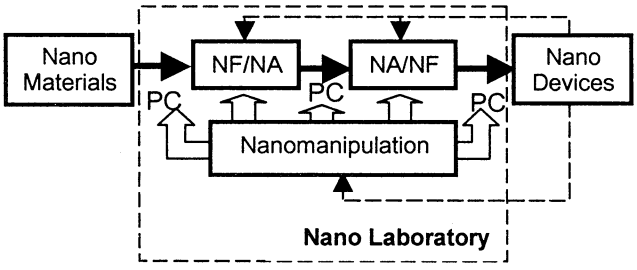


Fig.1 Hybrid Strategy for Nano Manufacturing
(PC: Property Characterization, NF: Nano Fabrication, NA: Nano Assembly)

2. Nano Laboratory

In our system, carbon nanotubes (CNTs) have been used as the main materials because of their exceptional properties (briefly summarized in Table 1) and broad potential applications [1]. In bulk state, nanotubes can be used to synthesize conductive and high-strength composites, to fabricate field emission devices (flat display, lamp, gas discharge tube, x-ray source, microwave generator, etc.), to save and convert electrochemical energy (supercapacitor, battery cathode, electromechanical actuator, etc.), to store hydrogen, and so on. However, the most promising applications of nanotubes that have deepest implications for molecular nanotechnology need to maneuver the tubes individually to build complex nanodevices. Such devices include nanoelectronics and nano electromechanical systems (NEMS) (concisely listed in Table 2).

Almost all of such applications and many unlisted potential ones of nanotubes for nanoelectronics and NEMS involve charactering, placing, deforming, modifying, and/or connecting nanotubes. Although chemical synthesis may provide a way for patterned structure of nanotubes in large-scale, self-assembly may generate better regular structures, we still lack of capability to construct irregular complex nanotube devices. Nanomanipulation [2], especially nanorobotic manipulation[3-7], with its "bottom

up" nature, is the most promising way for this purpose.

Table 1 Property of CNTs

Property	Item	Data
Geometrical	Layers	Single-walled nanotubes (SWNTs) or Multiwalled nanotubes (MWNTs)
	Aspect Ratio	10-1000
	Diameter	~0.4nm to >3nm (SWNTs) ~1.4 to >100nm (MWNTs)
	Length	Several μm (Rope up to cm)
Mechanical	Young's Modulus	~1 TPa (steel: 0.2TPa)
	Tensile Strength	45GPa (steel: 2GPa)
	Density	1.33~1.4g/cm^3 (Al: 2.7 g/cm^3)
Electronic	Conductivity	Metallic/Semi-conductivity
	Current Carrying Capacity	~1TA/cm^3 (Cu: 1GA/cm^3)
	Field Emission	Activate Phosphorus at 1~3V
Thermal	Heat Transmission	>3kW/mK (Diamond: 2kW/mK)

Table 2 Applications of CNTs in Nano Devices

Device	Fabrication
Diode	Rectifying diode: a kink junction.
Transistor	Room-temperature (RT) field-effect transistors (FETs): a semiconducting SWNT placed between two electrodes (source and drain) with a submerged gate RT single electron transistor (SETs): a short (~20 nm) nanotube section.
ICs	Hybrid logic circuits: two nanotube transistors placed on lithographically fabricated electrodes:. Pure nanotube circuits: interconnected nanotubes (intermolecular and intramolecular junctions.
Switch	Pushing and releasing a suspended nanotube [8].
Memory	Electromechanical nonvolatile memory: suspended cross-junctions [9].
Probe	Manually assembly, chemical vapor deposition (CVD), controlled assembly, and picking up a tube from vertically aligned SWNTs. [10]
Tweezers	Assemble two CNT bundles on a glass fiber [11].
Scissors	Nanorobotic assembly and shape modification [12].
Sensor	Chemical sensor: a semiconducting SWNTs.
Bearing	Open a MWNT by electric pulses [13].

A nano laboratory is presented as shown in Fig.2, which consists of a nanorobotic manipulation system (Fig. 3), an instrumentation system (a filed emission scanning electron microscope (FESEM) and a conventional atomic force microscope cantilever or a piezolever), and a nanofabrication system based on electron-beam-induced deposition (EBID) or manipulations. The specifications of the nano laboratory are listed in Table 3. The nano laboratory can be applied for manipulating nano materials—mainly but not limited to CNTs, fabricating nano building blocks, assembling nano devices, *in situ* analyzing the properties of such materials, building blocks and devices. The functions of it are summarized in Table 4, and many have been demonstrated elsewhere as shown in the references in the table.

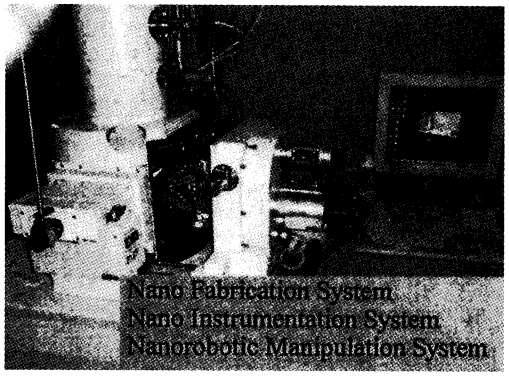


Fig.2 Nano Laboratory

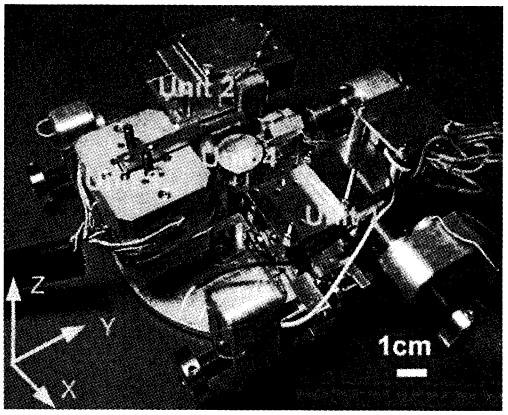


Fig.3 Nanorobotic Manipulators

Table 3 Specifications of Nano Laboratory

Item	Specification
Nanorobotic Manipulation System	
DOFs	Total: 16 DOFs Unit1: 3 DOFs (x, y and β; coarse) Unit2: 1 DOF (z; coarse), 3-DOF (x, y and z; fine) Unit3: 6 DOFs (x, y, z, α, β, γ; ultrafine) Unit 4: 3 DOFs (z, α, β; fine)
Actuators	4 Picomotors™ (Units 1& 2) 9 PZTs (Units 2& 3) 7 Nanomotors™ (Units 2 & 4)
End-effectors	3 AFM cantilevers+1 substrate or 4 AFM cantilevers
Working space	18mmx18mmx12mmx360° (coarse, fine), 26μmx22μmx35μm (ultrafine)
Positioning resolution	30nm(coarse), 2mrad (coarse), 2nm(fine), sub-nm (ultrafine)
Nano Instrumentation System	
FESEM	Imaging resolution: 1.5 nm
AFM Cantilever	Stiffness constant: 0.03nN/nm
Piezolever	Stain gauge built-in
Nanofabrication System	
EBID	FESEM emitter: T-FE, CNT emitter, Gas introduction system

Table 4 Functions of Nano Laboratory

Function	Involved Manipulations
Property Characterization	Mechanical properties: buckling [5] or stretching
	Electric properties: placing between two probes (electrodes)
Nanofabrication	EBID with a CNT emitter and parallel EBID [14]
	Destructive fabrication: breaking [15]
	Shape modification: deforming by bending and buckling, and fixing with EBID [15]
Nanoassembly	Picking up by controlling intermolecular and surface forces, and forces [7]
	Connecting with van der Waals [15]
	Soldering with EBID [15]
	Bonding through mechanochemical systhesis [16]

3. Nanotube Devices

3.1 Nanotube Mass Flow Sensors

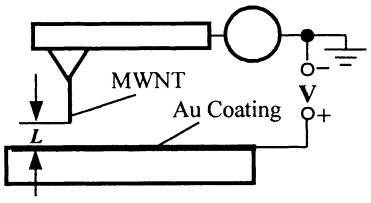


Fig.4 Design of a Mass Flow Sensor

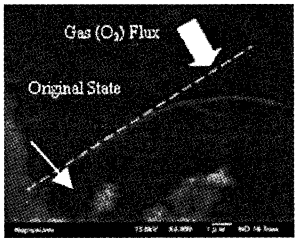


Fig.5 A Mass Flow Sensor

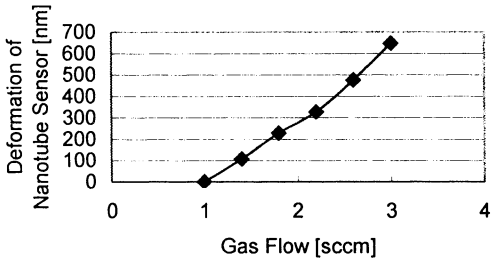


Fig.6 Calibration of Nanotube Mass Flow Sensor

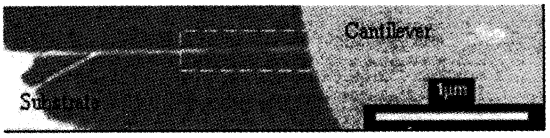
Measurement of ultra small flux of gases is an important and challenge problem. Silicon based microelectromechanical system (MEMS) can provide cantilevered thin wire as the transducer. But it is still difficult to fabricate vertical Silicon beam for measuring mass flow as small as several sccm. Like a nanotube pN force sensor, a cantilevered nanotube with very large aspect ratio is also possible to be used as a transducer for ultrasmall gas flow.

Fig.4 shows a design. By measuring the field emission current or tunneling current (as the gap L<~1nm), it is possible to detect the deflection of the nanotube caused by gas flow. Fig.5 shows a cantilevered nanotube bundle (length is about 10μm). When a flow of O_2 gas comes on it, deflections appeared. The relations between the mass flow (in sccm) and the deflection of the nanotube is shown in Fig.6. It can be found that the nanotube mass flow sensor is quite sensitive. Resolution is 0.93×10^{-3} sccm/nm.

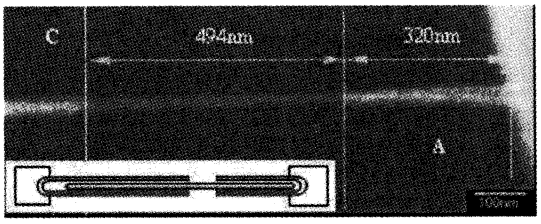
3.2 Nanotube Linear Bearings

Through destructive fabrications, typically, a layered structure and a sharpened structure can be obtained [15].

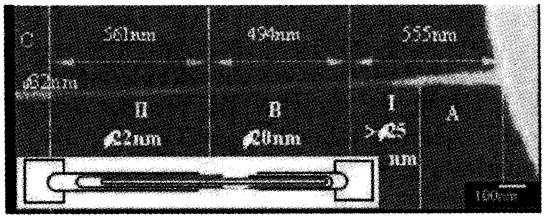
As shown in Fig.7 (a), a MWNT is supported between a substrate (left end) and an AFM cantilever (right end), which are in turn fixed on the Units 1 and 2 of the manipulation system. Fig.7(b) shows a zoomed up image of the central blocked part of Fig.7(a), and the inset in it shows its structure schematically. It can be found that the nanotube has a thinner neck (part B in Fig. 7(b)) that was formed by destructive fabrication, i.e., by moving the cantilever to the right. To move it more in the same direction, a motion like a linear bearing is observed as shown in Fig.7(c), and it is shown schematically by the inset.



(a)



(b)



(c)

Fig.7 Nanotube Linear Bearing

By comparing Fig.7(b) and (c), we can find that part B remained unchanged in its length and diameter, while its two ends brought out two new parts I and II from parts A and B, respectively. Part II seems have uniform diameter (ϕ22nm), while part I is a tapered structure with the smallest diameter ϕ25nm. The interlayer friction has been predicted very small [13], but the direct measurement of the friction remains a challenging problem.

3.3 Nanotube Scissors

Nanotube scissors are designed for probing the conductivity or cutting a nanotube between the two arms with saturated current. A new design by applying conductive EBID deposits is shown in Fig.8. To improve the cutting accuracy, it is necessary to modify the opening between the two arms (g). The key technique is the shape modification of nanotubes.

Fig.9 shows nanotube scissors fabricated by assembling two tubes on a commercially available AFM cantilever. Before shape modification, the opening between the two arms is larger than 1μm, whereas it is 85.6nm after modification.

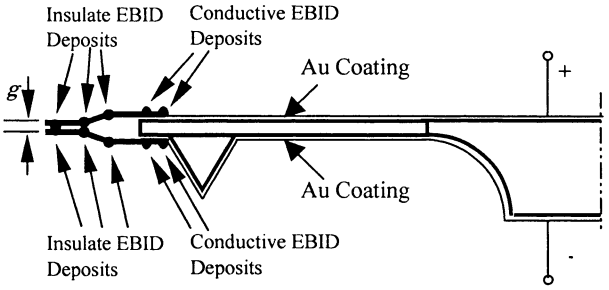
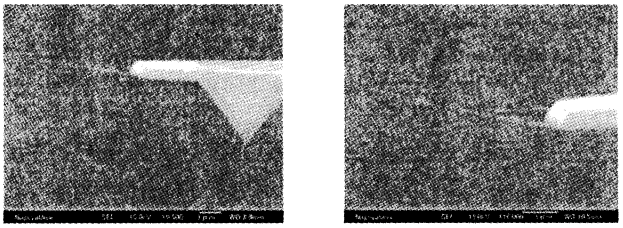


Fig.8 Nanotube Scissors



(a) Before Shape Modification (b) After Shape Modification

Fig.9 Nanotube Scissors (Scale bars: 1μm)

4. CONCLUSIONS

Based on a robotic manipulation system, a nano laboratory—a prototype nano manufacturing system has been presented, which is composed with a nanorobotic manipulation system with 4 units and 16 DOFs, a nano fabrication system based on EBID with an internal or external precursor evaporation reservoir equipped with a thermal field emission electron source or a nanotube cold cathode, and a real-time observation and measurement system based on FESEM equipped with 3-4 conventional AFM cantilevers, piezoresistive levers or nanotube probes. Nanotube devices including a mass flow sensor, a linear bearing, and nanotube scissors are fabricated in the nano laboratory.

Acknowledgments

This work was supported in part by the Scientific Research Fund of the Ministry of Education of Japan. We are grateful to Prof. Y. Saito at Mie University for providing us with MWNT samples.

References

[1] R.H. Baughman, A.A. Zakhidov, W.A. de Heer, "Carbon nanotubes --the route toward applications", *Science*, vol.297, pp.787-792, 2002.

[2] D.M. Eigler and E.K. Schweizer, "Positioning Single Atoms with a Scanning Tunneling Microscope", *Nature*, vol.344, pp.524-526, 1990.

[3] M. F Yu, M. J. Dyer, G. D. Skidmore, H. W. Rohrs, X. K. Lu, K. D. Ausman, J. R. Von Ehr, and R. S. Ruoff, "Three-dimensional manipulation of carbon nanotubes under a scanning electron microscope", *Nanotechnology*, vol.10, pp.244-252, 1999.

[4] L.X. Dong, F. Arai and T. Fukuda, "3D nanorobotic manipulation of nano-order objects inside SEM", in *Proc. of the 2000 International Symposium on Micromechatronics and Human Science*, Nagoya, Japan, 2000, pp.151-156.

[5] L.X. Dong, F. Arai, and T. Fukuda, "3-D nanorobotic manipulations of nanometer scale objects", *J. of Robotics and Mechatronics*, vol.13, pp.146-153, 2001.

[6] L.X. Dong, F. Arai, and T. Fukuda, "3D nanorobotic manipulations of multi-walled carbon nanotubes," in *Proc. of the 2001 IEEE International Conf. on Robotics and Automation (ICRA2001)*, Seoul, Korea, 2001, pp.632-637.

[7] L.X. Dong, F. Arai, and T. Fukuda, "Three-dimensional nanorobotic manipulations of carbon nanotubes", *J. of Robotics and Mechatronics (JSME)*, vol.14, No.3, pp.245-252, 2002.

[8] T.M. Tombler, C.W. Zhou, L. Alexseyev, J. Kong, H.J. Dai, L. Liu, C.S. Jayanthi, M.J. Tang and S.Y. Wu, "Reversible electromechanical characteristics of carbon nanotubes under local-probe manipulation", *Nature*, vol.405, pp.769-772, 2000.

[9] T. Rueckes, K. Kim, E. Joselevich, G.Y. Treng, C. L. Cheung and C.M. Lieber, "Carbon nanotube-based nonvolatile random access memory for molecular computing science," *Science*, vol.289, pp.94-97, 2000.

[10] H.J. Dai, J.H. Hafner, A.G. Rinzler, D.T. Colbert and R.E. Smalley, "Nanotubes as nanoprobes in scanning probe microscopy," *Nature*, vol.384, pp.147-150, 1996.

[11] P. Kim, and C.M. Lieber, "Nanotube nanotweezers," *Science*, vol.286, pp.2148-2150, 1999.

[12] L.X. Dong, F. Arai, and T. Fukuda, "Shape modification of carbon nanotubes and its applications in nanotube scissors", in *Proc. of IEEE Int. Conf. on Nanotechnology (IEEE-NANO2002)*, Washington, U.S.A., Aug.26-28, 2002, pp.443-446.

[13] J. Cumings and A. Zettl, "Low-friction nanoscale linear bearing realized from multiwall carbon nanotubes," *Science*, vol.289, pp.602-604, 2000.

[14] L.X. Dong, F. Arai, and T. Fukuda, "Electron-beam-induced deposition with carbon nanotube emitters", *Appl. Phys. Lett.*, vol.81, pp.1919-1921, 2002.

[15] L.X. Dong, F. Arai, and T. Fukuda, "Three-dimensional nanoassembly of multi-walled carbon nanotubes through nanorobotic manipulations by using electron-beam-induced deposition," in *Proc. of the 1st IEEE Conf. of Nanotechnology (IEEE NANO2001)*, Maui, Hawaii, 2001, pp.93-98.

[16] L.X. Dong, F. Arai, and T. Fukuda, "3D nanoassembly of carbon nanotubes through nanorobotic manipulations", in *Proc. of the 2002 IEEE Int'l Conf. on Robotics & Automation (ICRA2002)*, Washington, U.S.A., May 11-15, 2002, pp.1477-1482.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A mode to create artificial intelligence
---From Knowledge to Intelligence

Y. G. Zhang
Institute of Systems Science, Academia Sinica, Beijing, China.
yzhang@mail.iss.ac.cn

M. Sugisaka
Dept. of Electrical and Electronic Engineering,
OitaUniversity, Japan

ABSTRACT In this paper authors show some heuristic thinking first, and then proposed an engineering definition of artificial intelligence in artificial brain that is the abilities to obtain knowledge, to use knowledge and to operate knowledge. There are still two open important problems: one is the expression of knowledge for this goal and the other is how to construct the integration up operation and detailed down operation.
Keywords: Artificial Intelligence, Creating intelligence, Knowledge, cognition, Evolution of intelligence.

I. Introduction

In recent years many scientists and engineers reconsider the problem that creating artificial intelligence in bottom-up way, it is really a critical study. In fact, we hope a more natural manner to create artificial intelligence, that means we would not impose some "intelligence" on a system, but this system could have some abilities of interaction with environment and learn knowledge from outside. **Here, we would like to describe an engineering definition that "Intelligence is the abilities to obtain knowledge, to use knowledge, and to operate knowledge".**

This definition maybe suitable for the research on artificial intelligence but is not philosophic. We do not want to argue with this definition except describing our thinking and explaining what we are going to do.

Some heuristic examples

Example 1. Wolf boy

There was a story happened in 1950's, people found a human boy in a cave wolves lived, this boy seemingly was four years old. However, he cannot speak and walk vertically, and his behavior just likes a wolf. People took him back and tried to educate him, certain progresses had made but he had not been having the same intelligence like other boys. Why? Because he was taught by wolf's mother in his babyhood, but he not receive human mother's teaching although he had a human brain.

Example 2. Deaf boy

If a boy were deaf, then he just learns from his mother's action to know how to walk, how to understand mother's emotion, he cannot learn speak. Thus, he cannot communicate with other people during his babyhood till he could learns dummy language. Even if he did he still cannot understand many abstract things and concept easily. He was not as clever as others. If a boy was dumb only, the situation was better then deaf boy. Obviously, part of intelligence is from understanding of natural language, especially, he learned from mother during his boyhood.

Example 3. The comparison of intelligence with both Reptile and Mammal

In the real nature the animals' intelligence is quite different, some of them, those are in a high evolutionary level, has higher intelligence than others. An adequate example is Reptile and Mammal. Usually, Mammal has higher intelligence then Reptile animal. Most of reptile mother born eggs and never teach her newborn baby how to survival. An exception is Crocodile. However, all mammal' mother must teach her newborn baby how to survival, the baby has abilities to obtain knowledge from mother, and use those knowledge to survival. Some experiments showed us that Primate has

* This work is partly supported by Chinese Natural Science Foundation, Number # 60275016

stronger ability to cognize a little complicated pictures and language voice. They could understand human's emotion and instruction.

Previous examples show that intelligence of animal presents as the abilities of obtaining knowledge and use knowledge. If they could operate knowledge then they will have higher intelligence. Usually, the Primate in Mammal class has higher intelligence then other Mammal animals because they could operate the knowledge they leaned from community and environment outside.

These examples inspired us many scientific thinking that if we want to create an artificial brain with primitive intelligence, it has to possesses the abilities to obtain knowledge, to use knowledge and to operate knowledge it learned.

II. Knowledge and cognition

To obtain knowledge means acceptation of sensitive signal and transfer it into 'information' by using of neural system, these information include vision, hearing, touching, smelling, taste and so on, all are the sensitive information. Knowledge is the understanding of sensitive "information" and storing it as "knowledge".

If we want to create intelligence we have to make a machine that has ability to obtain knowledge. That is we have to use computer vision and computer hearing equipments and so on, to transfer the accepted signals as 'information' and to understand them. Now, the technique of computer vision to understand the picture and understanding of natural language have taken great progress in both theory and practice.

A lot of cognition scientific experiments and psychological study for children tell us animals and human children are how to obtained knowledge, these are very important for creating intelligence. Here we want to emphasize mother's role. At beginning period of babyhood they just could see some objects surrounding them and could hear the voice from mother mainly. Mother's voice and gesture language make baby establish a mapping gradually —the concrete object and the mother's teaching voice named that object, first is some still objects, and then some moving object. Of course, not only vision and hearing. This mapping will store in baby's brain. When this mapping was established firmly, the baby had obtained knowledge.

To use knowledge usually means to retrieve the knowledge stored in brain and to make decision by using them. Brain does not retrieve the original signal, but the pressured signal—information and the understanding of that information---knowledge. When baby could answer your some simple question, such as that Where is your toy? What are glasses? Who is father? Etc. that means the baby could use knowledge he obtained before.

If we create artificial brain, and we say that it has artificial intelligence, that implies it could accept the sensitive information from its "eyes" and "ears", they could sense the exist of environment and could processing sensitive information. We try to play a "mother" and to teach it. This process should be cognitive, or the process of obtaining knowledge. It is from simple to complex gradually. A complicated problem is to understand natural language, we believe that language is a very powerful tool to develop intelligence. How does this artificial brain to answer some questions? In another words, could this artificial brain use the knowledge it obtained? It is seemingly not very difficult if we have a knowledge base designed perfectly. There are two important problems: One is what expression of knowledge is like? Another is how does this artificial brain to operate knowledge?

III. Operating Knowledge

To operate knowledge means integration up with knowledge obtained and detailed down with knowledge. 'Integration up operation' of knowledge implies to understand "knowledge" from concrete to abstract in macro as "concept". The 'Detailed down operation' implies to understand "knowledge" as description of details in micro. These two operations carried out simultaneously in a progress. This process is a creating process of artificial intelligence. However, this study is quite challenged.

We still illustrate this concept by baby's actions. Mother shows and tell baby father's glasses, late she shows grandfather's glasses, and she show him more about glasses after certain days. If mother show him a glasses never seen before, and ask baby what it is? the baby could answer her it is a glasses. Why?

Because baby has used description in detail down operation to understand what is their common attributes? Two legs and two round-shaped lens, it may ware it on ears and nose to read. The baby could also integrate them himself or be taught by 'mother', and then he has a 'new' knowledge about glasses to help him to identify glasses. Even more, if mother show him a sunglasses he can realize it is still glasses with different usage. During this process operating knowledge baby obtain some new integrated knowledge, it might be formed a concept of 'glasses'. A new circle to obtain knowledge to use knowledge and to operate knowledge will start and go on in a new process. (See Figure 1 and 2). Maybe we could say that this is "emergence"?

In this process there are two way to obtain knowledge, one is supervised learning by 'mother' and the other is unsupervised learning.

The operation of knowledge depends on concrete algorithms, some is clever and some one is not very excellent. It will result in the intelligence difference between various artificial brains.

IV. Expression of knowledge

Now we try to discuss the problem of expression of knowledge. General speaking, the expression of knowledge depends on the real problem and applications. Obviously, the knowledge set of a chemist is definitely different from that of a physician. However, they have accepted the same basic education, which is related to their especial studies. So, we can say that their knowledge have the same structure, at least similar one.

From the discussion above it is clearer that in babyhood, he/she just see the things in room and hear mother's voice. Gradually, he/she can understand meaning of some noun and then some verb. Finally, he/she can understand some simple sentences. Before this he/she can pronounce some words, and then some incomplete sentences. In this process baby's knowledge is from his/her vision and listening, and then is touching and smelling senses. In the same time his /her intelligence becomes higher and higher. We can say his/her knowledge consists of semantic of (still and moving) image and natural language. In addition, knowledge should have a tree like shaped structure and grow up gradually.

The structure of tree like shaped has very important feature, operating knowledge can be done on this structure: the results of integrated up and detailed down are still in the same structure. In fact, this structure possesses fractal attribute. The question is how to put knowledge into this structure? In another words, how to design knowledge base?

V. Intelligence is evolutionary results of knowledge

No doubt, the natural intelligence is an evolutionary result. 'God' did not embed it into brain. So, the creation of artificial intelligence should be evolvable. The definition of intelligence mentioned here emphasizes to operate knowledge, so knowledge created in artificial brain can be changed and modified. At beginning, intelligence in babyhood maybe very simple, for example, a three month old baby understands an apple is an almost round shape and red color, couple of months late he/she knows that apple can be eaten, and have sweet and sour flavor. That means the knowledge about apple in baby's knowledge base was changed and modified. In fact, knowledge is often changed and modified during the baby's growing period. Especially, the abilities to obtain knowledge, to use knowledge and to operate knowledge is getting stronger and stronger when baby grows up, that implies the intelligence is become higher. It absolutely is different from the 'expert systems' appeared in artificial intelligence research before. The previous structure is evolvable; but the old expert system is embedded. Essentially, the evolvability is from the ability to obtain knowledge, to use knowledge and to operate knowledge. Expert system can use some knowledge fixed in only, but cannot obtain new knowledge and cannot operate knowledge.

This evolvable intelligence could deal with some abstract as 'concept', and make the artificial brain communicate with environment outside.

Now our group is doing this research and complementing it by computer, years late we could show some experiment results.

REFERENCES

[1] Zhang,Y.G. and Sugisaka,M., "Might we train an

artificial brain as a baby?", The Proceedings of the seventh international symposium on artificial life and robotics (AROB'02), pp.92-97, Beppu, Japan,2002

[2] Zhang, Y.G. and Sugisaka. M., "KANSEI Technology and Artificial Brain", (invited talk) The Proceedings of International Symposium on KANSEI technology, Muroran Institute of Technology, Hokkaido, Japan, October 5-6, 2001

[3] Zhang, Y.G., Sugisaka, M. and Wu, X.Q., "Bottom-up Way To Development of Multi-Cellular Digital Organization", **Journal of Artificial Life and Robotics**, Springer(Tokyo), No. 3, Vol.5, 2001.

[4] Zhang, Y.G. and Sugisaka, M., "Trainable Artificial brain (TAB)", The Proceedings of International Symposium of Knowledge and Systems Science (KSS), Dalian, China, September, 2001

[5] Zhang, Y.G., Nagashima, T, and Shimohara, K. "Humanization of computer with KANSEI technology", **The Journal of Three Dimensional Image**, Vol.14, No.4, pp.76-80, Dec. 2000

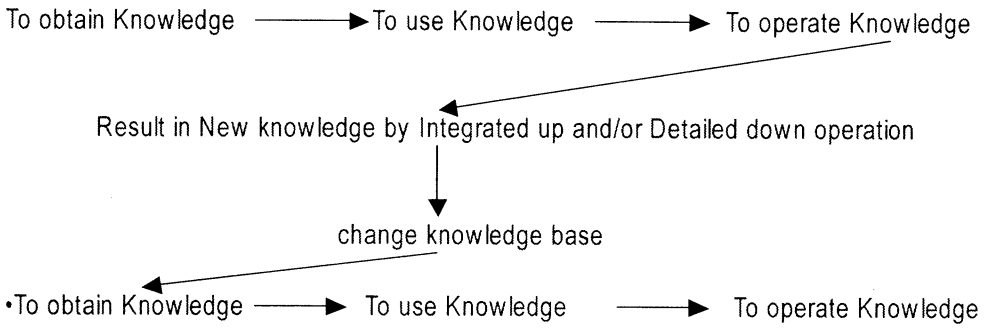


Fig.1

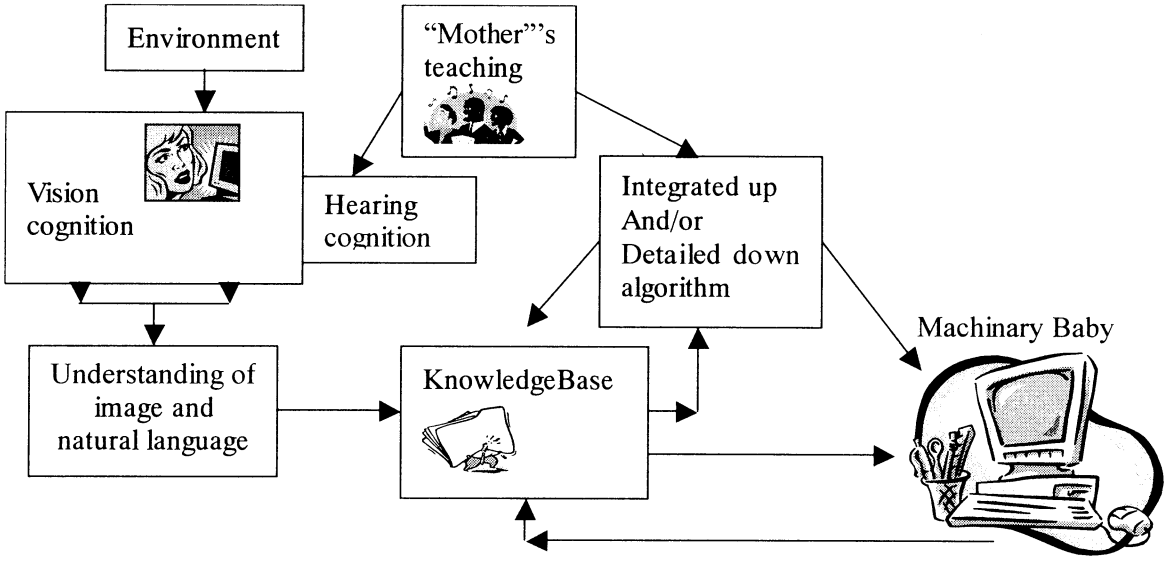


Fig. 2

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Robotics in the Emergence of Complexity Science

Jeffrey Johnson and Pejman Iravani
Department of Design and Innovation
The Open University
Milton Keynes, England, MK7 6AA
j.h.johnson@open.ac.uk

Abstract

Multi-robot systems possess most of the characteristic features of complexity. This alone means that robotics will play a role in the emerging science of complexity, if only as an example. However, robots have special features that make them especially suited to research into complexity science itself. These include the possibility of taking humans 'out of the loop', which is especially useful when investigating deep issues in representation, including the autonomous formation of constructs and vocabulary, and other issues related to culture in societies of autonomous agents. Robots can be used to conduct series of systematic experiments that are not feasible for many other classes of complex system. Also, robotics provides a unique laboratory for the 'can you trust it?' problem of simulation science. It is concluded that robotics will play a very important role in the developing science of complexity.

1. Introduction

Robotics is set to play a unique role in the evolution of the new science of complexity. There are many reasons for this, including

- multi-robot systems have all the characteristics of complexity
- robot systems can be studied without the complications of human agents
- robot systems can be used for systematic laboratory experiments
- robot systems can be both built and simulated, allowing simulation to be investigated as a scientific tool

Complexity science includes studying those systems that cannot be represented by systems of computationally reducible equations. This includes systems that are not obviously representable by systems of equations alone, such as medical care, manufacturing, businesses, and metropolitan planning. These systems require relational structure in their representation, often in multi-level lattice hierarchies. In this context, systems of equations may be part of the representation, but they are often very sensitive to initial conditions and are chaotic.

A further complication of complex systems is that they often have large numbers of variables, associated with the many autonomous heterogeneous 'agents' from whose interactions higher level system behaviour emerges. Associated with this, complex systems often have relational co-evolution between subsystems that is also very sensitive to initial conditions.

Multi-robot systems exhibit all these features of complexity, and more besides.

One of the most accessible examples of multi-robot systems is robot soccer. It is an excellent example because the system is easy to specify technically, including the ultimate goal of the system being, literally, to score more soccer goals that the opponent team of robots. The idea has been well implemented as a research challenge by the International RoboCup Federation, and also the Federation of International Robot-soccer Associations. These include definitions of various categories of physical and simulation robot leagues. Because of its accessibility and generality, robot soccer will be used as the example in this paper.

2. Complexity Science

The basic ideas of complexity science will just be sketched here. There is no agreed definition of what makes a system complex, but complexity science is concerned with systems that exhibit one or more of the following properties:

- system behaviour *emerges* from the *interactions* of individual agents:
  - autonomous subsystems can be considered to be *agents*
  - the number of agents may be *large but not huge* ($10^2 - 10^8$)
  - there is *combinatorial explosion* in the space of possible interactions
  - the *search* for optimal solutions is in large irregular spaces

- *representation* requires a *multi-level lattice hierarchical vocabulary*

  - representation is *explicit* inside computers
  - representation involves numbers and *n-ary* relations
  - complex systems may have *incomplete and inconsistent data*
  - there are *irreducible constructs*:
  - cannot vary the factors one at a time
  - there may be *system boundary* definition problems
  - the representation may be *self-similar* between levels

- computationally irreducible dynamics

  - *system time* is measured by *system events* in clock time
  - system behaviour is *sensitive to initial conditions* (chaotic)
  - agents are adaptive
  - there is *co-evolution* between subsystems
  - there is *co-evolution* of a system with its environment
  - system behaviour can be predicted by *simulation*
  - simulation may have the *can-you-trust-it?* problem.

2.1 Agents, interactions, and emergence

Human beings are examples of autonomous agents. Put ten million people together in London and anything can happen. Nonetheless, there are certain behaviours that must be predicted for administrative purposes, for example the number of people attempting to travel into, out of, and within the metropolis.

By comparison, complex behaviour can emerge from the interactions of just a few agents. For example, the thirty two chess pieces inhabiting a universe of sixty four squares.

Often complexity comes from the combinatorial explosion as the system trajectory evolves through time. If London's road system could take just one of two new states each week, in a year there are 2^{52} possible trajectories – billions! Of course the number of changes is much larger than two per week, as roads are dug up, bus services change, the weather changes, businesses open and close, and so on.

Agents may have simple rules of behaviour, but the interaction between many agents, in hundreds, thousands or millions, leads to a combinatorial explosion which makes a formulaic representation impractical.

2.2 Representation

Complexity science is computer-enabled, with methods that would be impractical without huge databases and powerful processing engines. This means that at least some part of the system must be represented explicitly inside computers.

This will be illustrated by a discussion of transportation in London, in order to illustrate the kinds of problems that robotics research can address. The London transportation system is already partly represented inside computers. However, much of the dynamics of the system is not understood, so only part of the system is explicit in this way.

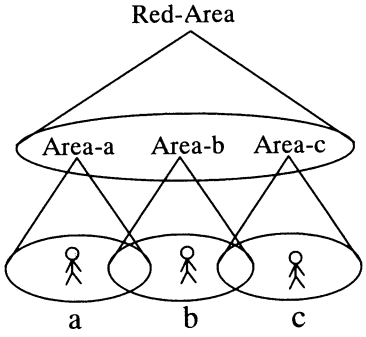
Generally the representation is multi-level. For example, in London the individual traveller is represented at a lower level of aggregation to the set of people travelling by bus. Similarly, houses are aggregated into zones, and zones are aggregated into higher level zones with 'London' being at the highest level.

The system of transportation in London involves many numbers, including census counts. It involves many relationships, including those between the agents and geographical locations. Most human activity takes place in geographical space, and the dynamics of the relationships between people and places can be very important.

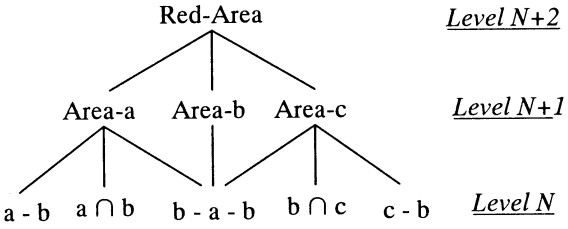
Many factors influence travel in London. These include availability of housing, availability of convenient public transport, availability of schools and public services, availability of shops and private services. These are interdependent, and trying to understand transport in terms of just one while keeping all the other fixed would give little insight into the dynamics of the system over the short or long term.

Where does the boundary of the London transport system lie? Does it include the commuters from Reading? Does it include Heath Row? Does it include Paris? Wherever it is drawn there is always a bit on the outside that appears as if it should be on the inside.

Conventional approaches to hierarchical structure involve tree structures in which the system can be increasingly divided into disjoint parts. This is not possible in complex systems. For example the set of train travellers and the set of bus traveller may have significant intersections. For this reason, complex systems generally require lattice hierarchical representations.



(a) Player areas aggregate into side's area



(b) lattice hierarchy of sub-areas

Figure 1. A lattice hierarchy

Figure 1(a) shows three players from a 'red team' of soccer players. Each player controls an area, Area-a, Area-b, and Area-c. It is perfectly natural that these areas intersect. This resulting classes of sets and subsets form a *lattice hierarchy*, as shown in Figure 6(b).

Within the lattice hierarchy, elements and structure are given *names*. These form the *vocabulary* of the system. Any construct that is not in the vocabulary cannot be operated on. The vocabulary spans the space of things that can be said about the system, and getting the vocabulary 'right' is essential in understanding complexity. However, the combinatorial nature of the system may allow different vocabularies. For example, a city may have its buildings aggregated into zones in different ways. Which is correct? The answer is that they may all be correct, in that when they are used consistently they allow useful things to be said about the system.

When humans devise a vocabulary to represent a system, we usually get most of it right and some of it wrong. This is why classification is so difficult, especially when attempting to do content analysis of unstructured texts, such as the answer to an open question in a survey. Automating the abstraction of vocabulary from data is one of the holy grails of complexity science.

Robotics has all these representation problems.

In robot soccer, *all* the representation is explicit in the controlling computers, or explicit in the physical embodiment of the robot. Generally the presentation is multilevel, using constructs provided by human builder. This leads immediately to the question as to how robots can abstract their own constructs by interactions with their environments. Humans sense the world differently to robots, and our physical interaction with the world is different to robots. Robots may abstract better vocabularies for their universe than can be designed by humans.

It is a question of the greatest importance in complexity science, since our computers will have to absorb vast quantities of heterogeneous information. Some of this will come in the form of free format natural language. Can computers scan such data to abstract its meaning in the given context? Can computers abstract appropriate constructs from the data they contain? If they can't, then there will always have to be a human 'in the loop' to make such data useful.

Construct abstract by robots explicitly does not have 'humans in the loop', and allows the possibility of new languages evolving to represent particular systems. Perhaps many of the systems we consider to be complex feel that way because humans have not formulated the necessary constructs that could help us to understand them better?

2.3 Dynamics

Until recently, science has been characterised by a particular kind of mathematical representation enabling *computational reducibility*.

For example, Halley used the new methods of Newton to predict correctly the next occurrence of a famous comet. His method did not just predict the next occurrence of Halley's Comet, but *all* future occurrences of Halley's Comet. To compute the n^{th} occurrence of

Halley's Comet it is not necessary to predict the $n\text{-}1^{th}$ occurrence, the whole process can be *reduced to one computation*.

The great advantage of computationally reducible representations is that they give classes of predictions. Input the numbers for a particular case, and output the prediction for that case. Most of the systems considered to be complex do not admit computationally reducible representations with the system dynamics made predictable by solving systems of equations.

The discovery of chaos in the mid-twentieth century required scientists to rethink completely what is meant by the term 'prediction'. Chaotic systems are characterised by them being bounded, and being sensitive to initial conditions. *Bounded* means that none of the variables representing the system go off to infinity in theory, or that the system does not disintegrate in practice. *Sensitivity to initial conditions* means that small changes in initial conditions result in big changes over time. Initial conditions are expressed as measurements, measurements can only be made to a finite level of accuracy, so *all* measurements have errors. In chaotic systems these undetectable errors cause rapid divergence in system behaviour.

Road traffic systems are known to exhibit chaotic behaviour at the microlevel, causing the shockwaves that are so often encountered on motorways. Transportation sub-systems exhibit co-evolution between subsystems, such as the cars, buses, trams, trains, subways, and so on. Also transportation subsystems co-evolve with their environment, include the business subsystem, the retail subsystem, the entertainment subsystem, and so on.

Most complex systems have their own internal dynamics, determined by their own events. These events occur in clock time, but it may not be possible to give point predictions of them in clock time.

For example, the event that a car park is full can cause congestion in an area. Depending on other factors, the car park may remain full for a long or short period in clock time.

Irrespective of this, the logic of the system is that *if* the car park if full *then* there will be local congestion. Then another rule may come into play: *if* there is congestions at A then it will be transmitted to B. Although it may not be possible to predict precisely in clock time when the car park will be full, it may be possible to predict the consequences in system time.

As another example, a bridge may be scheduled to be built an opened on a given day. If the bridge is finished early, it may be put into services early. Certainly, if it is finished late, it will be put into service late in clock time. However, as far as the system is concerned, the important event is the bridge being finished. This leads to the tautology that when the event of the bridge being finished has occurred, then the bridge can be put into service.

For many complex systems, *simulation* is the only way to investigate their dynamics. Usually this involves establishing transition rules for the interaction of the individual agents, with system behaviour emerging bottom-up as these are iteratively applied.

For most systems, a single simulation conveys little information, since most complex systems are sensitive to initial conditions. The contribution of simulations is to provide distributions of possible outcomes that give insights into the system dynamics.

Some simulations start with small data sets and generate large data sets. Some simulations start with massive data sets and generate even more massive data sets. For example, the TRANSIMS system developed at Los Alamos National Laboratory uses large population data sets to generate synthetic populations of millions of people in geographical space, and simulate their trip making by various transportation modes. In doing so it generates terabytes of intermediate data.

There are many issues associated with simulation. One of the most troubling is the 'can you trust it problem' [1]. It is not difficult to find simulations that you would not trust. However, for a simulation that one wants to trust, there is at present no test to see if a simulation is 'correct', and for some simulations the question may be undecidable.

3. Multi-robot systems & complexity science

Multi-robot systems have many of the characteristics of complexity, and the emerging science of complexity is highly relevant in their design. On the other hand, complexity science has many unanswered questions that robotics is ideally placed to answer. The most important of these include

the possibility of doing systematic and replicable experiments, and human beings being 'out of the loop' in data collection, data processing and pattern recognition, optimisation and planning, and execution.

There are many kinds of robots, and many kinds of multi-robot systems. To make the ideas concrete, the example of robot soccer will be used in this section. Robot soccer players may be small wheeled robots playing with a golf ball, or larger wheeled robots playing with a soccer ball. Research into humanoid soccer robots is at a relatively early stage. The stated objective of the International RoboCup Federation is to have a team of humanoid robots beat the human world champions by 2050.

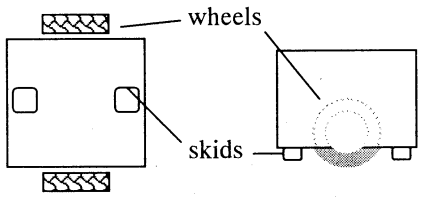
3.1 Multi-robot systems exhibit all the features of complexity science

The kind of autonomous mobile robots that play soccer can be considered to be *agents*, and the behaviour of a team of robots emerges from their interactions. There are twenty two robot agents in the system, and the ball, and the pitch environment. There is a combinatorially astronomic space of possible configurations of robots, and many aspects of robot soccer use search techniques.

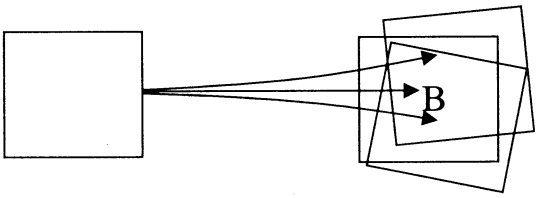
The representation for robot soccer requires a multi-level lattice hierarchical vocabulary, with areas of the pitch defined as zones, and configurations of robots being recognised and named structures. The representation is explicit inside computers, and it involves numbers and n-ary relations between players and the pitch. The system has incomplete and inconsistent data, since the number of sensors is limited by practical considerations, and the interpretation or pattern recognition of the fused data has a non-zero rate of errors. The system has irreducible constructs and it is not possible to vary the factors one at a time.

System boundary definition problems are less severe in robot soccer than in other systems, and this makes them a very good laboratory subject. One aspect of robots is that their universes are obviously spanned by their sensors. In some sense, the information boundary of a robot is defined by its sensors. Investigation of this could inform the question of boundary definition in other complex systems.

Robot soccer certainly has computationally irreducible dynamics with system time being measured by system events in clock time.



(a) a simple two-wheeled robot



(b) same starting position, different end positions

Figure 1. Most robot systems are very sensitive to initial conditions

Even the simplest robots have motion that is sensitive to initial conditions. For example, consider a square robot with two independently driven wheels programmed to move from A to B on a flat surface. Then start the robot from A and you will observe a resulting position and orientation. Start the robot again from A, and it will almost certainly end up at B' $\neq$ B. Most machines are like this, and that is why they require *controllers*.

The teams in robot soccer co-evolve as they play each other. To date much of the co-evolution has involved hardware improvements, making individual robots better controlled in their environment.

There has also been considerable co-evolution in the simulation leagues, as the human programmers learn from each other.

Simulation is a particularly important feature of robot soccer. In robotics it is common to test programs by graphical simulators, and it is common to find them very useful in the first level of debugging programs. A misplaced minus sign in a trigonometric expression is easy to detect in simulation when the behaviour is obviously wrong. In robotics simulation allows systems to be investigated before they are built. In this sense the simulations are predictions of the real system.

From the perspective of complexity science this is very interesting. Robotics is one of the few examples of complex system that can both be simulated and built. In other words, robotics is the perfect laboratory for the 'can you trust it problem'.

3.2 Robot systems can be studied without the complications of human agents

Researchers and analysts of many human systems are often 'stakeholders' in the outcome of the research. This can lead to a lack of scientific objectivity. One of the astonishing characteristics of humans is the ability to formulate predictions on the basis of no prior explicit theory or data: we can encounter a new environment, formulate an implicit *ad-hoc* model, and make relatively good decisions on how to act in that environment. Often this involves making analogies between diverse domains, and requires a great deal of knowledge in many areas. Many large complex systems are managed on the basis of intuition, implicit data, implicit theory, and inspired decision making. Eliciting this information can be a complicating factor in trying to understand human systems.

By comparison, robot systems can be studied in very objective ways. Data comes from objective sensors, and can be recorded. The theories on which robots operate are either built in, or explicitly the subject of research. These theories are in very limited domains and are much simpler than the kind of theories that humans formulate.

Complex human systems can be difficult to study because the people in the system have their own interests. Also humans can be very opinionated about things, and we are not always the best instruments for collecting data. By comparison all observations in robot systems are made by explicit sensors and all data in robot systems is explicit within the representation. This makes the study of robotic systems much more objective than most studies of human systems.

Human beings have wonderful pattern recognition abilities that far surpass anything that can be done by machines. Machine vision is a simple example in which humans excel and machine, including robots, perform poorly. The same applies to more abstract patterns. Much of what happens in the human brain is implicit. When we see a bird we don't have to think about it. When a human soccer player sees his team mates move in a certain way, he does not need to cogitate on what to do next. It's obvious! Humans make similar obvious and implicit decisions about many other things, including how to micromanage a business meeting, or how to micromanage an organisation. Systems that use implicit human pattern recognition are hard to investigate.

By comparison, in robot systems all pattern recognition in robot systems is automatic, making them much easier to interrogate and understand.

Another complicating factor in studying human systems is the social structures that act as a backcloth for information flows. In robot systems all information flows depend on explicit connectivities, making their analysis much easier.

3.3 Robot system can be used for systematic laboratory experiments

It is not possible to do systematic experiments in many human systems. For example, voting one political party out of power is a social experiment that precludes the possibility of leaving that party in power. Similarly, making a change to the regulation of the Internet is an experiment that precludes the possibility of observing the effects of not making that change. Apart from this, there are situations in which it would be too dangerous, expensive, or unethical to conduct systematic experiments. For example, there is considerable opposition to the current international experiment testing the proposition that unrestricted carbon emissions have no adverse effect on the atmosphere.

By comparison, you can have it both ways in robotics and run the system with and without the change, or across a wide range of possibilities.

With robots it is possible conduct systematic series of experiments to investigate evolution and co-evolution between robots and their environment.

Representation is a major research area in complexity science, and the abstraction of appropriate vocabularies is an important issue. Robot systems can be used to investigate automatic language formation, individual and group learning, as in the 'talking heads' experiments [2,3,4]. It has been suggested that robot systems can be used to investigate artificial cultures, as constructs emerge and

become integrated into individual and group vocabularies[5].

Complexity science is largely motivated by applications, especially the possibility of engineering emergence. The ability to plan experiments with robot system promises significant insights into this.

5. Conclusion

Having given some of the characteristics of complex systems, it has been argued robots exhibit them all to a greater or lesser extent. This alone means that robotics will play a role in the emerging science of complexity.

However, robots have special features that make them especially suited to research into complexity science itself. These include the possibility of taking humans 'out of the loop', and conducting series of systematic experiments that are not feasible for many other classes of complex system. Robotics promises to give powerful insights into the problem of construct and language formulation for the study and control of complex systems. It also provides an ideal laboratory to study the 'can you trust it?' problem of computer simulation. In summary:

- **multi-robot systems have all the characteristics of complexity:**

  - autonomous robots can be considered to be agents
  - system behaviour emerges from the interactions of individual robots
  - system behaviour can be predicted by simulation
  - system behaviour is sensitive to initial conditions
  - robot systems have incomplete and inconsistent data
  - representation requires a multi-level lattice hierarchical vocabulary
  - system representation involves *n-ary* relations
  - irreducible constructs: cannot vary the factors one at a time
  - co-evolution between subsystems co-evolve, with the environment
  - robot systems are computationally irreducible
  - combinatorial explosion and search as a problem-solving paradigm

- **robot systems can be studied without the complications of human agents**

  - all observations in robot systems are made by explicit sensors
  - all data in robot systems is explicit within the representation
  - all pattern recognition in robot systems is automatic
  - all information flows depend on explicit connectivities

- **robot system can be used for systematic laboratory experiments**

  - robot system can be both real and simulated, providing a unique laboratory for the 'can you trust it problem' of simulation.
  - robot systems can be used to investigate evolution and co-evolution robot system can be used to investigate automatic language formation
  - robot systems can be used to investigate individual and group learning
  - robot systems can be used to investigate artificial cultures
  - robot systems can be used to research engineering emergence

My conclusion from this discussion is that robotics is set to play a very important role in the developing science of complexity.

References:

[1] Johnson, J.H., 'The "Can You Trust It?" Problem of Simulation Science in the Design of Socio-technical Systems', *Complexity,* **6**(2), 34-40, 2001

[2] Marocco, D., Cangelosi, A., Nolfi, S., 'The role of social and cognitive factors in the emergence of communication: experiements in evolutionary robotics, *Phil. Trans. Royal Soc. London* - A(2003).

[3] Steels, L., 'Intelligence with representation', *Phil. Trans. R. Soc. Lond.* A (2003) 361, 2381-2395, 2003

[4] Steels, L., 'Evolving grounded communication for robots', Trends in Cognitive Sciences, **7**(7), July 2003

[5] Johnson, J.H., Sugisaka, M. 'Culture and communication in the design of swarm robot control systems, Int. Conf. Advanced Systems Engineering, Coventry University, 2000.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A Navigation Scheme for Mobile Robots Using Global and Local Path Planning

Gi-Hyung Park
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
gihyung2@daum.net

Jin-Hyun Park
Dept. of Mechatronic Eng.
Jinju National Univ.
Jinju, 660-758, Korea
jh-park@jinju.ac.kr

Young-Kiu Choi
School of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
ykichoi@pusan.ac.kr

Abstract

In this paper, the path planning and tracking problem are mentioned to guarantee efficient and safe navigation of AMR(autonomous mobile robot). We focus on the global path planning, and we also deal with local path planning to avoid unexpected obstacles. In the global path planning, we improved the DT(distance transform) algorithm to have shorter paths than those obtained using the conventional DT algorithm. In the local path planning, the navigation algorithm consists of a path tracking behavior and an obstacle avoider behavior. The behaviors are constructed using the fuzzy system. Simulation studies have been carried out to show the effectiveness of the proposed navigation algorithm scheme.

Keywords : Path planning, Path tracking, DT algorithm, Mobile robot

1. Introduction

Most of the existing navigation methods for indoor environments use two main types of planning: global path planning and local path planning. Global path planning usually involves strategies that generate hazard-free paths, based on known environmental map. Global path planning is able to produce optimal paths; however, it cannot cope with environments with unknown obstacles. On the other hand, local path planning works well in dynamic and unknown environments, But it does not guarantee optimal paths and is inefficient when the goal is far from AMR and the environment is cluttered. Hence, we combine the global path planning scheme with the local path planning to ensure the optimal path with collision avoidance for unexpected obstacles.

This paper is composed of two main parts which are the global path planning and local path planning. In the stage of the global path planning, we propose an improved DT algorithm. The conventional DT algorithm generates paths that consist of horizontal lines, vertical lines, and slants with fixed angle[1][2]. Using this algorithm, it is impossible to generate the minimum-distance paths. But the improved DT algorithm generates paths composed of lines with unrestricted slopes; therefore, it can obtain shorter-distance paths compared to the conventional DT algorithm. In the stage of the local path planning, AMR tracks the preplanned path using path tracking behavior based on the fuzzy inference system. And an obstacle avoidance behavior based on the fuzzy inference system is added to avoid unexpected obstacles on the preplanned paths.

To prove the effectiveness of the proposed algorithm, performance studies for the path planning and tracking are carried out through computer simulations.

2. Environment representation

Environment-representative method for robot system is determined based on the robot hardware capability such as sensor system or computational system. Most of the existing environment-representative methods for indoor environment use environment representation based on grid-cells or models[3].

2.1 Environment representation based on grid-cells

Environment-representative method based on grid cells assumes workspace to be a plane of two dimensions and allots a certain value representing yes or no of obstacle existence to all the grid cells after representing workspace as triangular, rectangle cells and so on. Therefore, using this method, we can represent workspace as grid map composed of two dimensional arrays and are easy to represent workspace comparing to other environment-representative methods because obstacle characteristics are not considered. But this method has a defect that memory space for grid map increases very fast when the size of the workspace is increased.

2.2 Environment representation based on model

Environment-representative method based on models is method that saves the characteristic parameters after extracting characteristics from obstacles[4]. This method can be easily applied in the workspace having big or simple shape obstacle, but it has a difficulty of extracting the characteristic parameters in complex workspace.

2.3 Environment representation for experiment

In this paper, we use the environment-representative method based on grid cells. Even though this method has a defect that memory space for saving gird map increases with workspace size, it has merits which is easy to be applied and represent obstacles. Therefore, this method is suitable for indoor environments.

3. Global path planning

Global path planning is a method that generates paths after checking the entire workspace with the known environment. In this paper, an improved DT algorithm for global path planning is used to generate paths. This algorithm enables AMR to generate paths regardless of shape or size of obstacles. Compared to the conventional DT algorithm, this algorithm makes shorter paths.

3.1 Distance transform algorithm

DT algorithm is a method that generates path after making potential map from target grid cell to start grid cell[5]. At this time, we assume that AMR has environmental information.

The method that generates path using DT algorithm is as follows. First, we assign zero for the potential value of target grid-cell. 8-neighbor grid-cells of target grid-cell have one for the potential value. The same as those does, we assign a potential value form the target grid-cell to all the free space grid-cells. And grid-cells occupied by obstacle have a specific potential value Second, using the steepest descent method from the start grid-cell to target grid-cell we can find path. Figure 1 shows path generated using DT algorithm.

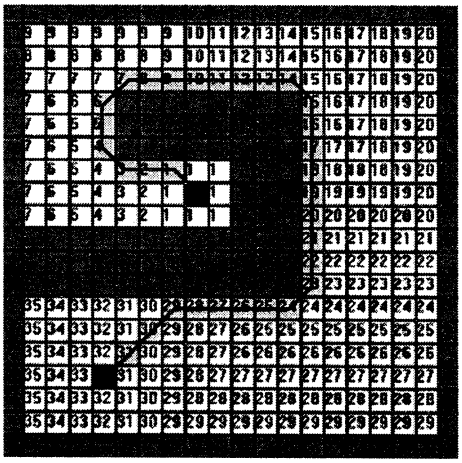


Fig. 1. Application example of DT algorithm

3.2 Improved distance transform algorithm

The conventional DT algorithm generates paths that consist of horizontal lines, vertical lines, and slants with fixed angle. Using this algorithm, it is impossible to generate the minimum-distance paths. But the improved DT algorithm generates paths composed of lines with unrestricted slopes; therefore, it can obtain shorter-distance paths compared to the conventional DT algorithm. Also existing DT algorithm has a defect that is difficult to decide the searching order about 8-neighbor. Therefore, In this paper Improved DT algorithm is proposed. This algorithm has merits which simplifies the searching order through changing potential value assignment and searching order about 8-neighbor into those about 4-neighbor, and shorten paths compared conventional DT algorithm through path re-generation process.

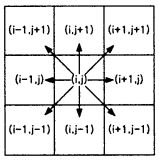


Fig. 2. 8-neighbor search

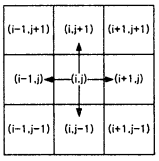


Fig.3. 4-neighbor search

Path re-generation process is as follows. Paths are consist of several path lines. Path re-generation process has two processes that amend path lines that are not fit and cancel path lines that are not necessary. In first process, path lines are generated as Figure. 6.

step0. set the path lines of n number : $L_1, L_2, ..., L_n$.
set the path grid - cells of m number
: $C_1, C_2, ..., C_m$ in order from start grid - cell.

step1. $i = 1, k = 1$.
start - point of L_1 ← center - point of C_i.
end - point of L_1 ← center - point of C_2.

step2. $i = i + 1$
if($C_i == C_{goal}$)
{ Goto step4}
if obstacle isn't detected on line consists of
start - point of L_k and center - point of C_i
{ Goto step2}

step3. refer the fig. 4, fig. 5.
renew i.
end - point of L_k = start - point of L_{k+1} = center - point of C_i.
$k = k + 1$.
Goto step2.

step4. end - point of L_k = center - point of C_i.
End.

Steps mentioned above explain the process of amending path lines. After all steps, dash line becomes solid line in figure. 6.

Figure 4 shows the case that wall-type obstacle grid-cell detected. When 4-neighbors of obstacle grid-cell are checked, if potential values of 3-neighbors equal to potential value of obstacle grid-cell, AMR recognize obstacle as wall-type. And end-point of k-th path line and start-point of k+1-th path line become center-point of renewed i-th path cell.

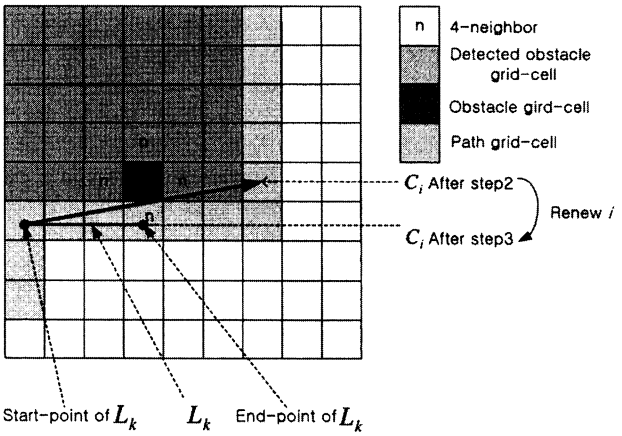


Fig. 4. Case that wall-type obstacle grid-cell detected

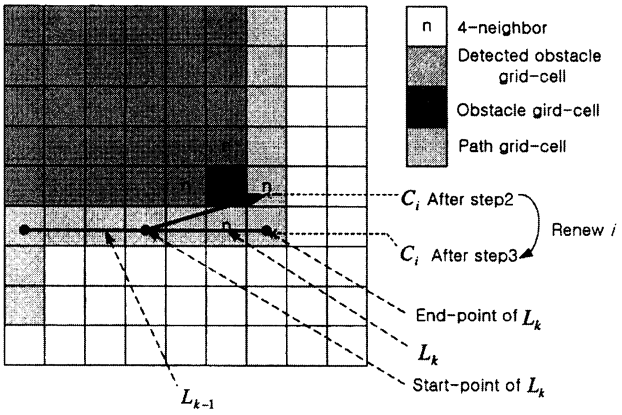


Fig. 5. Case that corner-type obstacle grid-cell detected

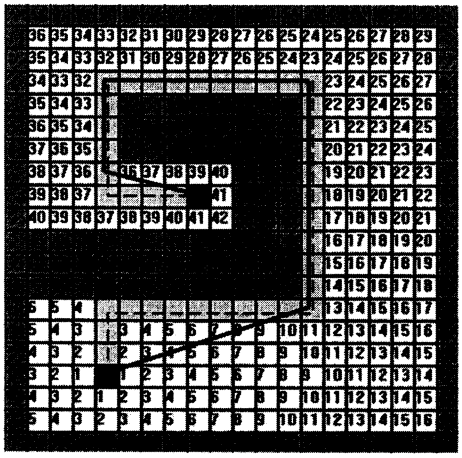


Fig. 6. Application example of improved DT algorithm

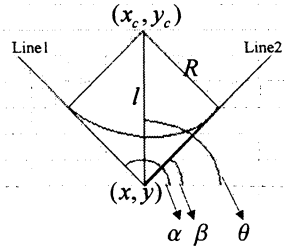
Figure 5 shows the case that corner-type obstacle grid-cell detected. When 4-neighbor of obstacle grid-cell are checked if potential values of 2-neighbor equal to potential value of obstacle grid-cell, AMR recognize obstacle as corner-type. And end-point of k-th path line and start-point of k+1-th path line become center-point of renewed i-th path cell.

If there are path lines that are not necessary, these path lines are cancelled in second process of path re-generation.

4. Local path planning and tracking

4.1 Circular path generation

AMR must navigate after stopping and turning in turning point of AMR, that is, joint point of two path lines. So this navigation makes entire navigation time long and paths having turning point isn't flexible paths. Therefore, in this paper, we make paths flexible through generating circular arc path in turning point. Figure 7 shows circular arc generation.



l = Diameter of AMR (1)

$$\theta = \frac{(\alpha + \beta)}{2} \quad (2)$$

$$x_c = x + l\sin\theta \quad (3)$$

$$y_c = y + l\cos\theta \quad (4)$$

$$R = l\sin(\theta - \alpha) \quad (5)$$

Fig . 7. Circular arc generation

Figure 8 shows simulation example of circular arc generation. Hazard-free zone is considered.

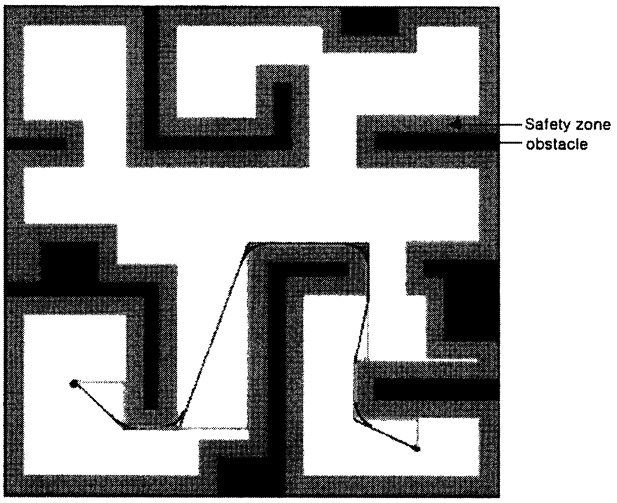


Fig. 8. Simulation example of circular arc generation

4.2 Path tracking behavior

In this paper, path tracking behavior based on the fuzzy inference system is used to track the preplanned paths. Fuzzy input parameters are the distance and angle between AMR and path.

4.3 Obstacle avoidance behavior

When the obstacles are detected by ultra sonic sensors, obstacle avoidance behavior makes AMR navigate following the wall of the obstacle. After avoiding the

obstacle completely, AMR resumes to track the preplanned path using the path tracking behavior.

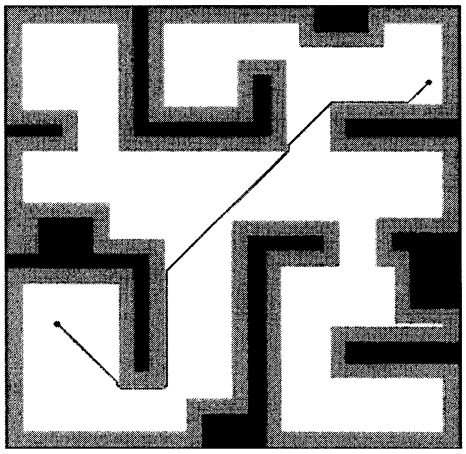


Fig. 9. Simulation of path planning using DT algorithm(path distance: 1956cm)

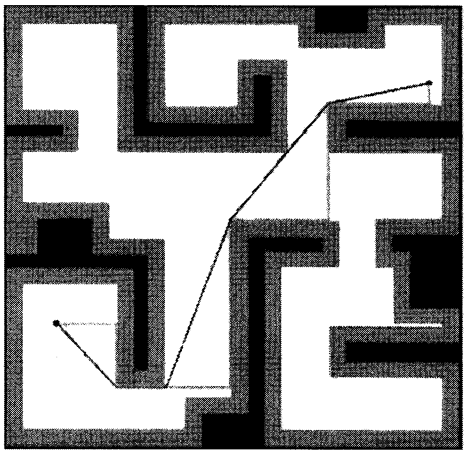


Fig. 10. Simulation of path planning using improved DT algorithm(path distance: 1882cm)

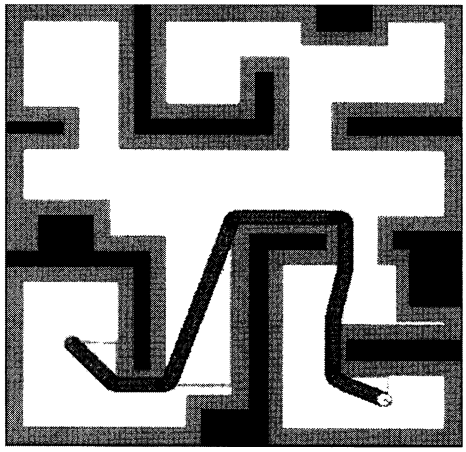


Fig. 11. Case without curved arc paths (path distance:2067cm, navigation time:45sec)

5. Results of computer simulation

A grid-cell size for experiment is 10cm × 10cm. And size of workspace is 15m × 15m. Figure 9,10 shows paths generated using DT algorithm and using improved it respectively. Improved DT algorithm makes shorter path than existing DT algorithm. Figure 11,12 shows that path tracking behavior is well-operating and case having curved arc path is better than case without those.

6. Conclusions

Using the improved DT algorithm, it is demonstrated that we can improve the defect of the existing DT algorithm and shorten path distance.

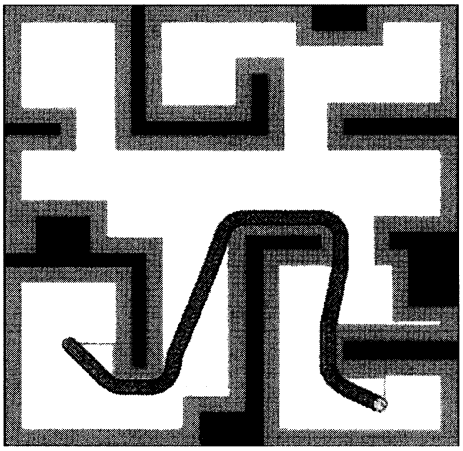


Fig. 12. Case having curved arc paths (path distance:1991, navigation time:42sec)

References

[1] Joon Seop Oh, Jin Bae Park and Yoon Ho Choi, " Complete coverage navigation of clean robot based on triangular cell map," IEEE International Symposium on Industrial Electronics, vol. 3, pp.2089 - 2093, June 2001

[2] Lim Chee Wang, Lim Ser Yong and Ang M.H. Jr, " Hybrid of global path planning and local navigation implemented on a mobile robot in indoor environment, " Proceedings of the 2002 IEEE International Symposium on Intelligent Control, pp.821 – 826, Oct. 2002

[3] H. Choset and J. Burdick, " Sensor Based Planning, Part I : Generalized Voronoi Graph," IEEE International Conference on Robotics and Autonomous, pp.1643-1648, 1995

[4] Tomas Lozano-Perez, "Spatial Planning : A Configuration Space Approach," Proceedings of the Trans. on Computers, vol. 32, no. 2, Feb. 1983

[5] A. Zelinsky, " A Mobile Robot Exploration Algorithm," *IEEE Trans. on Robotics and Automation*, Vol. 8, No. 6, pp. 707-717, Dec. 1992

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Robust Acceleration Control of Wheeled Mobile Robot Moving on Rough Surface

Makoto WADA, Masahiro OYA, Hideki HONDA, Toshihiro KOBAYASHI and Hujio OHKAWA
Department of Mechanical and Control Engineering
Kyushu Institute of Technology, Tobata, Kitakyushu, 804-8550, Japan
E-mail: a584202m@tobata.isc.kyutech.ac.jp

Abstract

When wheeled mobile robots run at high speed on the rough surface, the devices loaded on the body are affected by the body acceleration and may be destroyed. In this paper, we propose a method to reduce the body acceleration of wheeled mobile robots. In this mobile robot, a ball screw is adopted as a mechanism to drive wheels up and down. To achieve small body acceleration, an ideal model is designed based on the state space description containing body acceleration. The information from accelerometers mounted on each wheel axis is used in the ideal model. Based on the designed ideal model, a robust model tracking controller is proposed. It is shown by carrying out experiments that the control performance is maintained even if the body weight is changed.

1. Introduction

In extreme environment, for example stricken area and planet surface, there are many dangerous tasks for human workers. In recent year, mobile robot working instead of human in the environment has been developed[1 – 6]. Mobile robot developed until now classify roughly into three large categories based on its mechanism, such as wheeled mobile robot[1 – 3], articulated body mobile robot[4, 5], and legged mobile robot[6]. Wheeled mobile robot with the high-speed mobility has an advantage in the case when it is required to gather information and carry devices as soon as possible. In the developed wheeled mobile robots[1 – 3], the mechanism to climb up a step, and to prevent carrying devices from falling down is mainly considered. In the case when these mobile robot run on rough surface at high speed, the devices loaded on the body are affected by the body acceleration and may be destroyed. In such a case, it is necessary to reduce body acceleration. However, in previous researches, a method to reduce body acceleration isn't considered.

In this research, we propose a method to reduce the body acceleration of wheeled mobile robot. A ball screw system is employed as a mechanism to drive wheels up and down. To achieve small body acceleration, an ideal model is designed based on the state space description containing body acceleration so that H_∞ norm of transfer function from rough surface to body acceleration becomes small. The information from accelerometers mounted on each wheel axis is used in the ideal model. Next, robust model tracking controller is developed so that ball screw displacement of mobile robot track that of ideal model. It is theoretically shown that the proposed controller doesn't require exact values of system parameters. It is also shown that we can make norm of tracking error arrive to any small closed region by setting only one design parameter. When small ball screw displacement tracking error is realized, it is expected that small body acceleration is also achieved. Moreover, practical running test results are given to show the effectiveness of proposed controller.

Fig.1 Mobile robot

2. Wheeled mobile robot and Passive robot

2.1 Wheeled mobile robot

The picture and the illustration of developed wheeled mobile robot are shown in Fig.1 and Fig2 respectively. In Fig.2, in order to simplify the following explanation for wheeled mobile robot, each wheel is numbered. The developed wheeled mobile robot is driven by wheels of No.2 and No.3 and ball screws are adopted as a mechanism to drive four wheels up and down. The illustration of a ball screw mechanism is shown in Fig.3. As shown in Fig.3, torque produced by the motor is transmitted through belt to the ball screw. Then the ball screw rotates and each wheel is driven up and down. In Fig.2, z is vertical displacement of center of gravity, z_r is vertical displacement of center of body, ψ is pitch angle of body, ϕ is roll angle of body, V is traveling velocity of body. z_i $(i = 1, 4)$ is vertical displacement on each point P_i. w_i is vertical displacement of rough surface on each wheel. Vertical acceleration of rough surface

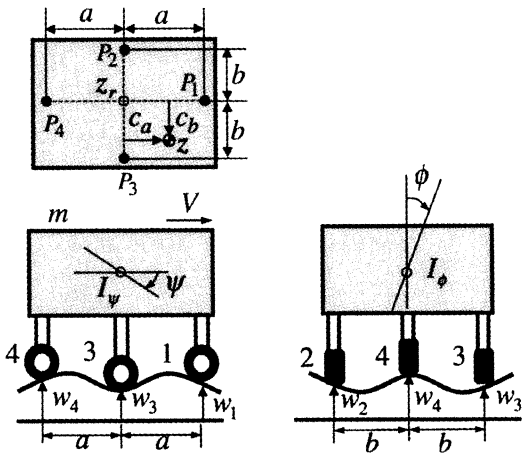


Fig.2 Illustration of mobile robot

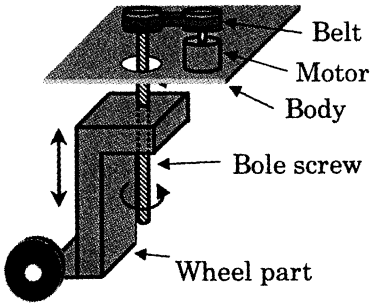


Fig.3 Bole screw mechanism

on each wheel is given by accelerometers mounted on each wheel axis. In order to confirm the validity of proposed controller, three accelerometers are mounted on body to measure body acceleration. The displacement of ball screw $x_i = z_i - w_i$ is measured by encoder mounted on motors. The velocity of ball screw is given by Eular approximation of ball screw displacement $\dot{x}_i \approx [x_i(kT) - x_i((k-1)T)]/T$($T$: Sampling time).

If it is assumed that the change of pitch angle and roll angle are small, dynamic equation of wheeled mobile robot is given by

$$M_q\ddot{\boldsymbol{q}}(t) + C_q\dot{\boldsymbol{q}}(t) = B_q\boldsymbol{u}_q(t) + D_{q1}\dot{\boldsymbol{w}}(t) + D_{q2}\ddot{\boldsymbol{w}}(t) - M_a\boldsymbol{g}_q, \quad (1)$$

$$\left.\begin{array}{l} \boldsymbol{q}(t) = [z \quad \psi \quad \phi]^T \\ \boldsymbol{u}_q(t) = [u_1(t) \quad u_2(t) \quad u_3(t) \quad u_4(t)]^T \\ \boldsymbol{w}(t) = [w_1(t) \quad w_2(t) \quad w_3(t) \quad w_4(t)]^T \end{array}\right\}, \quad (2)$$

$$\left.\begin{array}{l} H_h = H^TH,\ B_q = k_aH^T, C_q = d_aH_h, \\ M_q = M_a + k_l^2J_aH_h \\ D_{q1} = d_aH^T, D_{q2} = k_l^2J_aH^T \\ H = \begin{bmatrix} 1 & -(a-c_a) & c_b \\ 1 & c_a & b+c_b \\ 1 & c_a & -(b-c_b) \\ 1 & a+c_a & c_b \end{bmatrix} \\ M_a = \mathrm{diag}\{m, I_\psi, I_\phi\},\ \boldsymbol{g}_q = [g \quad 0 \quad 0]^T \end{array}\right\}, \quad (3)$$

Table 1 Nominal values of mobile robot

m	Vehicle mass	39.8kg
a	Distance between center of vehicle and front/rear axle	0.208m
b	Distance between center of vehicle and right/left axle	0.235m
c_a,c_b	Distance between C.G. and center of vehicle	0.0m
J_a	Moment of ineatia of motor ball-screw system	5.95×10^{-6} kgm^2
I_ψ	Pitch axis moment of ineatia	0.553kgm^2
I_ϕ	Roll axis moment of ineatia	0.395kgm^2
d_a	Viscous damping constant	175Nms/rad
k_a	Electromechanical transmission constant between voltage and torque	108N/V
k_l	Transmission constant between rotational and linear motion	1.05×10^3 rad/m

where $u_i(t)$ is input voltage added to motors mounted on body (see fig.3). To simplify arguments, it is assumed that viscosity coefficient of each wheel axis are the same and constant value of characterizing the electromechanical conversion between current and torque of each wheel axis are also the same. The nominal parameters of wheeled mobile robot are shown in Table 1.

2.2 Passive robot

To keep the height of body at initial height, we consider the following control input $u_q(t)$

$$\boldsymbol{u}_q(t) = -k_f\boldsymbol{x}(t) - (\overline{k_l}^2\overline{J}_a/\overline{k}_a)\ddot{\boldsymbol{w}}(t) + \boldsymbol{u}(t) + \boldsymbol{r}, \quad (4)$$

where $\boldsymbol{x}(t) = [x_1(t) \quad x_2(t) \quad x_3(t) \quad x_4(t)]^T$ is measurable ball screw displacement vector, $k_f = 60$ is feedback gain, $\boldsymbol{u}(t)$ is new control input signal. $\boldsymbol{r} = [1.1 \quad 0.83 \quad 0.83 \quad 0.75]$ is constant input voltage to keep body horizontally. In this paper, wheeled mobile robot applied control input (4) is named passive robot. In the case of $\boldsymbol{u}(t) = 0$ and $\ddot{\boldsymbol{w}}(t) = 0$, the static displacement of body is defined by $\overline{\boldsymbol{\eta}}_p$. Introducing a new state vector $\boldsymbol{\eta}(t) = \boldsymbol{\eta}_p(t) - \overline{\boldsymbol{\eta}}_p$, the dynamic equation of passive robot can be expressed as

$$M_p\ddot{\boldsymbol{\eta}}(t) + C_p\dot{\boldsymbol{\eta}}(t) + K_p\boldsymbol{\eta}(t) = B_p\boldsymbol{u}(t) - D_p\ddot{\boldsymbol{w}}(t), \quad (5)$$

$$\left.\begin{array}{l} H_b = \overline{H}^TH, K_p = k_ak_f\overline{H}_h^{-1} \\ M_p = (H_b^{-1})^T(M_a + k_l^2J_aH_h)H_b^{-1} \\ C_p = d_a\overline{H}_h^{-1}, \quad B_p = k_a(H_b^{-1})^TH^T \\ D_p = (H_b^{-1})^T\left(M_aH_b^{-1}\overline{H}^T + \dfrac{\overline{k_l}^2\overline{J}_a}{\overline{k}_a}k_aH^T\right) \end{array}\right\}. \quad (6)$$

3.Robust tracking controller

To reduce body acceleration, we consider the state

vector $\zeta(t) = [\eta(t) \quad \dot{\eta}(t) \quad \overline{H}^T\overline{H}\ddot{q}(t)]$, including body acceleration vector $\ddot{q}(t) = [\ddot{z}(t) \quad \ddot{\psi}(t) \quad \ddot{\phi}(t)]$. According to dynamic equation (6), nominal system with only nominal parameters is given by

$$\left.\begin{array}{l}\dot{\zeta}(t) = A\zeta(t) + Bu_\zeta(t) + D\ddot{w}(t) \\ u_\zeta(t) = \dot{u}(t) + u(t)\end{array}\right\}. \tag{7}$$

Then, the following control is applied

$$u_\zeta(t) = -\frac{1}{2\mu}B^T P\zeta(t), \ \mu > 0, \tag{8}$$

where, P is a positive definite, symmetric solution of the Riccati Equation

$$\left.\begin{array}{l}P(A+\delta I) + (A+\delta I)^T P \\ \quad -P(BB^T/\mu - DD^T)P = -Q^TQ - \epsilon I \\ Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & Q_q(\overline{H}^T\overline{H})^{-1} \end{bmatrix}\end{array}\right\}, \tag{9}$$

$$\left.\begin{array}{l}Q_q = \mathrm{diag}\{10^{-2}, 10^{-1}, 10^{-1}\} \\ \delta = 0.3, \quad \mu = 5 \times 10^{-7}\end{array}\right\}, \tag{10}$$

where $\epsilon > 0$ and Q_q is weight matrix. δ is design parameter introduced to improve convergent rate. By using δ, we guarantee that all eigenvalues in closed system become smaller than $-\delta$. Then, H_∞ norm of transfer function from rough surface acceleration $\ddot{w}(t)$ to body acceleration $Q_q\ddot{q}$ becomes smaller than 1. If the value of weight for vertical acceleration of body center of gravity is set to very large value, it is difficult to run over a step. Therefore, it is necessary to set the value of weight for vertical acceleration of body center of gravity smaller than that of pitch and roll acceleration of body. By using (7) and (8), the ideal model is designed. Next, robust model tracking controller is developed so that ball screw displacement of mobile robot $\eta(t)$ track that of ideal model $\eta_d(t) = [I \quad 0 \quad 0]\zeta_d(t)$. Define tracking error as follows

$$\left.\begin{array}{l}s(t) = \beta^{-1}\dot{\tilde{\eta}}(t) + \tilde{\eta}(t) \\ \tilde{\eta}(t) = \eta(t) - \eta_d(t)\end{array}\right\}, \tag{11}$$

where β is design parameter introduced to improve tracking performance. Then, error system is obtained as

$$\begin{aligned}M_p\dot{s}(t) = {} & \beta M_p s(t) - \beta M_p\tilde{\eta}(t) - \beta^{-1}K_p\tilde{\eta}(t) - C_p s(t) \\ & + C_p\tilde{\eta}(t) + \beta^{-1}k_a u_a(t) - \beta^{-1}\sigma(t), \qquad (12)\end{aligned}$$

$$\sigma(t) = M_p\ddot{\eta}_d(t) + C_p\dot{\eta}_d(t) + K_p\eta_d(t) + D_p\ddot{w}(t). \tag{13}$$

If error signal $s(t)$ become small, and then, it is expected that body acceleration become small. This will be ascertained later by experiments.

To develop robust model tracking controller, the following assumptions are made for the system (12).

*A*1. There exists a bounded positive constant value $\overline{\rho}_m$ for any vector x such that $x^T(\overline{M}_p^{-\frac{1}{2}})^T M_p\overline{M}_p^{-\frac{1}{2}} x \le \overline{\rho}_m\|x\|^2$.

*A*2. There exists a bounded positive constant value $\overline{\rho}_c$ for any vector x such that $x^T(\overline{M}_p^{-\frac{1}{2}})^T C_p\overline{M}_p^{-\frac{1}{2}} x \le \overline{\rho}_c\|x\|^2$.

*A*3. There is a bounded positive constant value $\underline{\rho}_{ka}$ such that $|k_a| \ge \underline{\rho}_{ka}$.

*A*4. There is a bounded positive constant value $\overline{\rho}_{\tilde{\sigma}}$, such that $\|\tilde{\sigma}\| \le \overline{\rho}_{\tilde{\sigma}}$, where $\tilde{\sigma}(t)$ is error signal between $\sigma(t)$ and its estimate value $\hat{\sigma}(t)$ = $\overline{M}_p\ddot{\eta}_d(t) + \overline{C}_p\dot{\eta}_d(t) + \overline{K}_p\eta_d(t) + \overline{D}_p\ddot{w}(t)$, $\tilde{\sigma}(t) = (k_a/\underline{\rho}_{ka})\hat{\sigma}(t) - \sigma(t)$.

*A*5. There is a bounded positive constant value $\underline{\rho}_{mp}$ such that $\|M_p\| \ge \underline{\rho}_{mp} > 0$.

In error system (12), even when $M_p, K_p, C_p.B_p$ are uncertain matrices, if M_p and B_p are symmetric positive definite matrix so that it is easy to design a controller, $s(t)$ becomes stable. M_p is symmetric positive definite matrix but B_p is not symmetric positive definite matrix in (12). Therefore, to force B_p to be symmetric positive definite matrix, we consider the following controller.

$$u(t) = \overline{H}u_a(t), \tag{14}$$

$$u_a(t) = -\frac{1}{\underline{\rho}_{ka}}\left(\beta f_s\overline{M}_p s(t) - \hat{\sigma}(t)\right). \tag{15}$$

Then, the following theorem is shown. Before showing this, we define the positive definite function

$$\begin{aligned}V(t) = {} & s^T(t)M_p s(t) + \tilde{\eta}(t)^T M_p\tilde{\eta}(t) \\ & + \beta^{-2}\tilde{\eta}(t)^T K_p\tilde{\eta}(t) + \beta^{-1}\tilde{\eta}^T(t)C_p\tilde{\eta}(t). \qquad (16)\end{aligned}$$

Theorem. 1

If feedback gains f_s, f_{s1}, f_{s2} are set so as to satisfy the following relation

$$\left.\begin{array}{l}f_s = \beta f_{s1} + f_{s2} \\ f_{s1} \ge \dfrac{3\overline{\rho}_m + 1}{2}, \quad f_{s2} \ge \overline{\rho}_c\end{array}\right\}, \tag{17}$$

then, the closed loop system becomes stable and the tracking error $\tilde{\eta}$ satisfies

$$\|\tilde{\eta}(t)\|^2 \le \frac{\overline{\rho}_{\tilde{\sigma}}^2\|\overline{M}_p^{-1}\|}{\beta^4\underline{\rho}_{mp}}, \quad \|\dot{\tilde{\eta}}(t)\|^2 \le \frac{2\overline{\rho}_{\tilde{\sigma}}^2\|\overline{M}_p^{-1}\|}{\beta^2\underline{\rho}_{mp}}. \tag{18}$$

From Theorem 1, it can be concluded that the tracking performance can be improved by using only one design parameter β. Moreover, even if the body weight is changed, the control performance is maintained as long as assumptions [A1]-[A5] are satisfied .

Proof

To save space, only an outline of the proof is shown below. Considering (12),(14),(15) and relation

$$\left.\begin{array}{l}4s(t)^T C_p\tilde{\eta}(t) \le \tilde{\eta}(t)^T C_p\tilde{\eta}(t) + 4s(t)^T C_p s(t) \\ 2\beta^{-1}s(t)^T\tilde{\sigma}(t) \\ \quad \le \beta s(t)^T\overline{M}_p s(t) + \beta^{-3}\tilde{\sigma}(t)\overline{M}_p^{-1}\tilde{\sigma}(t)\end{array}\right\} \tag{19}$$

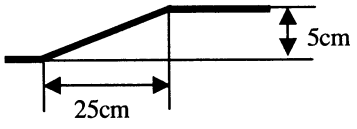


Fig.4 Rough surface

and analyzing the derivative of the positive definite function V(t), it is seen that the following inequality holds

$$\dot{V}(t) \leq -\beta V(t) + \beta^{-3}\overline{\rho}_{\tilde{\sigma}}^{2}\|\overline{M}_{p}^{-1}\| \qquad (20)$$

Form (20), it is seen that the closed loop system becomes stable. Then, the inequality (18) can be obtained.

4. Experiment

The proposed controller (14),(15) is applied to wheeled mobile robot shown in Fig.1. The velocity of mobile robot is set to 0.8 [m/s], sampling time is set to 0.5[ms]. The wheeled mobile robots run on rough surface shown in Fig.4. In general, it is difficult to measure position of body center of gravity. Therefore, vertical acceleration of center of body is used to evaluate vertical acceleration. Fig.5 shows body acceleration responses in mobile robot using proposed controller. Fig.6 shows body acceleration responses in passive robot. In Figs.5-8, (a) shows vertical acceleration response of center of body and (b) shows pitch acceleration response. The wheeled mobile robot reaches rough surface at 2.1 second.

As shown in Fig.5,6, it is seen that body acceleration of mobile robot using proposed controller become very small compared with passive mobile robot.

Next, to verify the robustness of the proposed controller, we carry out practical running test for mobile robot with different body weight. The actual value of the weight of body is set to 30.8 [kg]. Fig.7 shows body acceleration response in mobile robot with 30.8 [kg] using proposed controller. Fig.8 shows body acceleration response in passive robot with 30.8 [kg]. As shown in Fig.7,8, in passive robot, maximum of absolute value of pitch acceleration in the case of 30.8 [kg] increase by 40% compared with the case of 39.2 [kg]. On the other hand, in mobile robot with proposed controller, maximum of absolute value of pitch acceleration hardly changed. Therefore, it is seen that control performance is maintained even if body weight is changed.

5. Conclusion

In this research, we propose a method to reduce the body acceleration of wheeled mobile robot with a ball screw system. In practical running test, it is shown that body acceleration of mobile robot using proposed controller become very small compared with passive mobile robot. Moreover, it is also shown that the control performance is maintained even if body weight is changed. As a future research, proposed controller will be applied to stairs.

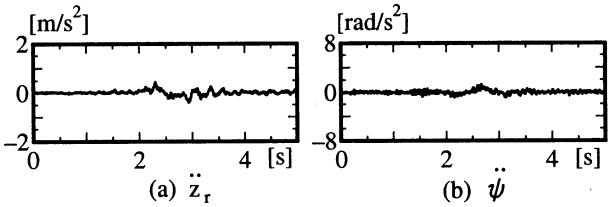


Fig.5 Mobile robot using proposed controller (m=39.2kg).

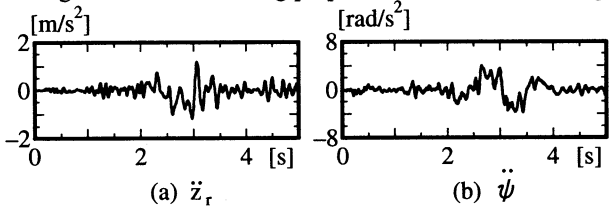


Fig.6 Passive robot (m=39.2kg).

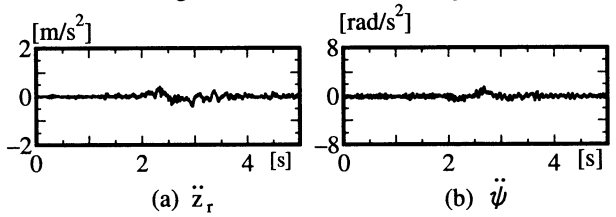


Fig.7 Mobile robot using proposed controller (m=30.8kg).

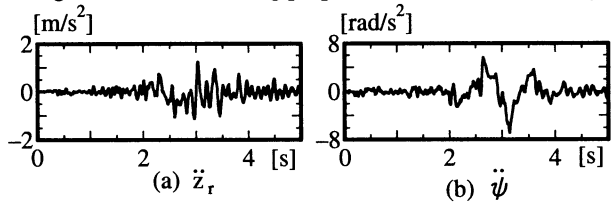


Fig.8 Passive robot (m=30.8kg).

Reference

[1] S.Hirose, M. Usa and S. Aoki, Development of Terrain Adaptive Ouadru-Wheel Vehicle HELIOS-Ⅲ, *Proc. of 9th Annu. Conf. of RSJ*, 1991, 305-306

[2] Y.Uchida, K.Furuichi, S.Hirose, Fundamental Performance of 6 Wheeled off-road Vehicle "HELIOS-V", *Proc. IEEE Int. Conf. Robotics & Automation*, 1999, 2336-2341

[3] K.Taguchi, Enhanced Wheel system for step climbing, *Advanced Robotics*, 1995, 137-147

[4] S. Hirose, E.F.Fukushima and S.Tsukagoshi, Basic Steering Control Method for The Articulated Body Mobile Robot, *IEEE*, 1994, 2384-2390

[5] M.Yim, D.G.Duff, and K.Roufas, PolyBot: a Modular Reconfigurable Robot,*Proc. IEEE Int. Conf. Robotics & Automation*, (2000), 514-52

[6] S. Hirose, K.Yoneda and H.Tsukagoshi, TITAN Ⅶ: Quadruped Walking and Manipulating Robot on a Steep Slope, *Proc. IEEE Int. Conf. Robotics & Automation*, 1997, 494-500

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

The Design of a Real-Time, Multiagent System for Controlling Distributed Devices in an Autonomous Robot System

Makoto Obayashi, Hiroyuki Nishiyama and Fumio Mizoguchi
Information Media Center
Tokyo University of Science
270-8510, Yamazaki 2669-1, Noda, Chiba, Japan.

Abstract

In this paper, we developed a real-time multiagent system for a robot control software system. This system is implemented by embedded computers and enables real-time execution and real-time communication in a distributed environment. Usually, an autonomous robot has many controlled actuators and sensors and needs many software components for making decisions. These components must be efficiently integrated in a robot system. However, there are problems in synchronization and shared information of interdependent software modules. Moreover, a real robot with actuators requires a real-time hardware system for implementing software servo systems and signal transmission. Considering these factors, we developed a real-time multiagent system that enables real-time communication, real-time processing, dynamic synchronization and dynamic elimination of competition in a distributed environment. Additionally, we adopted a real-time network interface to realize real-time communication between agents in different computers. Using our system incorporating these features, we constructed an autonomous robot system and demonstrated its ability through experiment results.

1 Introduction

In general, computers employed in robot systems should feature low-power consumption, small size and light weight. The same is also true of an autonomous mobile robot and a robot room equipped with various sensors and actuators. In particular, the power consumption of the computer inside an autonomous robot influences the amount of drive time. Reducing computer weight enables reducing the load applied to actuators supporting the robot. Furthermore, distributed control using a network is necessary for a robot room to satisfy extendibility and maintenance requirements. Considering these factors, embedded computers are suitable for constructing a computer system in an autonomous robot. The disadvantage of using an embedded computer is that it has lower throughput than a PC. However, it is possible to solve this problem by using many embedded computers. This idea is the same as in GRID computing in which many computers are connected with each other via a network to execute one task. Usually, an autonomous robot adopts a CPU bus connection architecture for system construction. For example, a VME bus can easily establish a real-time communication system between computers. In contrast, system construction using a network like TCP/IP has a major advantage in design flexibility. Unfortunately, applying this method to design a robot architecture has two problems. First, it requires synchronous processing and the elimination of competition in the distributed environment. Second, it is impossible to realize hard real-time networking using a TCP/IP network. An autonomous robot system requires sufficient response to the real environment and hard real-time for actuator control. Communication with actuators and sensors controlled by distributed computers via a network also requires hard real-time operation. This problem arises in both software and hardware and disturbs distribution of critical control processes that need hard real-time execution. It also requires a framework for integrating distributed processes.

2 System Design

In our approach to solve the problems described in the previous section, we implement a real-time multiagent system (RT-MRL). RT-MRL incorporates real-time execution in a multiagent language (MRL). It also has many features that differ from MRL[1], such as the following. First, we limit the execution environment of RT-MRL to embedded computers. Because an embedded computer is small and consumes little power, we consider that is a practical way to implement our system in an autonomous robot. Second, our system is executed in a distributed computing environment. We can thus use two or more embedded computers to compensate for the limited calculation capabil-

ity of a single embedded computer that is less capable than a PC. Third, we only use a special network interface that enables hard real-time communication between computers because usual network communication protocols such as TCP/IP can't realize hard real-time communication. We implement real-time communication between agents in a distributed computing environment via network infrastructure. We present an overview of RT-MRL, which has these features, in the next section.

2.1 Behavior Definition using RT-MRL

We developed a multiagent construction language to define each agent behavior that controls various devices and performs computational tasks. The status and behavior of an agent are changed by interference or events of other agents. An agent behavior can be represented as an automaton. The reaction of an agent to each specified event is then the state transition from the current state to another state. It is also necessary to implement exclusive operation and communication between unspecified agents and resolve competition among agents' outputs. Implementing these factors using a native computational language is very complex and requires much development labor. We solve this problem by developing a translator to translate from the agent behavior definition to the native language, enabling automatic creation of the various agent features described above. The prototype of the agent language is the MRL[1]. Its description form is an event-driven definition that enables simple definition of agent behavior. Additionally, RT-MRL developed in our system enables description of real-time execution. An RT-MRL program consists of predicates that show an agent's behavior response to a specified event. Its syntax is as follows.

$$\begin{array}{lcl} Head(arg0, arg1, ...) & :- & Guard0, Guard1, ... \\ & | & Body0, Body1, ... \end{array}$$

Here, $Head(arg0, arg1, ...)$, $Guard0, ...$, $Body0, ...$ represent predicates and their arguments, condition clauses, and execution clauses. Each predicate is distinguished by its name and argument. Predicates with the same name are identified as different predicates when they have different numbers of arguments. When an event occurs during agent processing, the system attempts to unify each predicate. The unified predicate, which has a suitable predicate name and arguments, is then executed. This enables simple description of an agent state transition. The following RT-MRL statement can define communication behavior between unspecified agents.

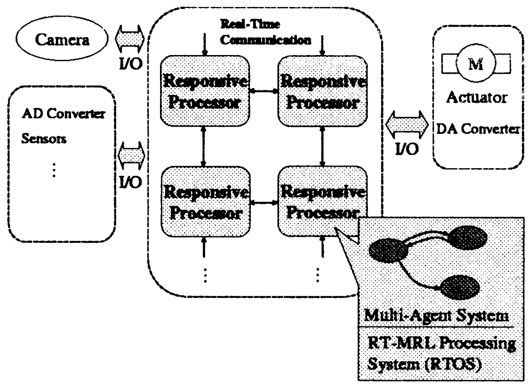


Figure 1: System configuration of physical network in RT-MRL

*<*Message*> %send or receive message to lower
^<*Message*> %send or receive message to upper

If the above description is declared in a Guard clause, it means receiving <*Message*>. If it is in the Body clause, it means sending <*Message*>.

3 Implementation

3.1 Execution Environment of RT-MRL

In our system, we have to implement hard real-time functions, which requires an implementation software servo system in the robot system. In particular, RT-MRL is developed to integrate asynchronous processes in the distributed computing environment. Therefore, a hard real-time function is required in both process execution and communication via a network. Hard real-time process execution is realized by adopting RTOS in our system. This problem is solved by constructing RTOS as the RT-MRL execution system as we described earlier. However, the hard real-time communication between computers cannot by realized by software methods. We therefore use a special network interface that enables hard real-time communication. First, ARCNET was a good candidate for the real-time network interface. ARCNET, in which the protocol is an improved token-passing method, can communicate within the maximum boundary time calculated in a specified formula. However, adopting ARCNET has some disadvantages in that the maximum boundary time increases according to the increase in the number of nodes and the communication speed is low. Considering these factors, we use the Responsive Processor[2] as the RT-MRL execution environment. The Responsive Processor has the Responsive Link, which enables realizing hard real-time communication. An RT-MRL execution environment is conceptually depicted in Fig. 1. A Responsive Link is connected peer-to-peer, and a Responsive Processor has four Responsive Links. The network configuration is represented in Fig. 1.

Each hardware component controlled by each agent is connected to each Responsive Processor's I/O. All packets are transmitted via several computers (except neighboring computers). This structure indicates that there is overhead in dynamic routing processing of a packet[3]. However, in the RT-MRL method, dynamic routing only occurs in the initial stage of establishing the P2P connection between agents. The communication delay due to packet routing does not affect real-time communication

3.2 Implementation of RT-MRL execution system

The software implementation of our system consists of two parts, implementation of the RT-MRL Translator and implementation of the RT-MRL execution system. The RT-MRL execution system was developed as part of the operating system for an embedded computer. Therefore, the RT-MRL execution system is the same as multitasking RTOS. Agent behaviors defined by RT-MRL language are translated to the native codes (C language) that become an RT-MRL application. Executable software is then created from the RT-MRL execution system and an application.

The MRL execution system assumes that the execution environment is limited to a parallel computer and that it can't execute on the distributed computing environment alone and has no real-time function. In contrast, our system adopts a Responsive Processor to realize hard real-time communication in a distributed computing environment. Therefore, it also enables real-time communication between agents in different computers. However, a different agent communication mechanism is required in a single computer than in several computers. To resolve this problem, we implement a message transmission mechanism developed by combining the message-passing method and shared-memory method. As a result, a developer can define the behavior of agent communication and can realize hard real-time communication without being concerned about differences in agent communication. Next, we describe the details of implementing the RT-MRL execution system.

The multiagent system in RT-MRL is composed of a tree structure that automatically creates a communication channel between an upper agent and a lower agent. Moreover, communication channels can be created dynamically between arbitrary agents. These channels are created by agent behavior and are managed in the shared memory of the RT-MRL execution system. Whenever a communication channel is created, an empty channel in the shared memory is assigned to it. The two agents trying to communicate

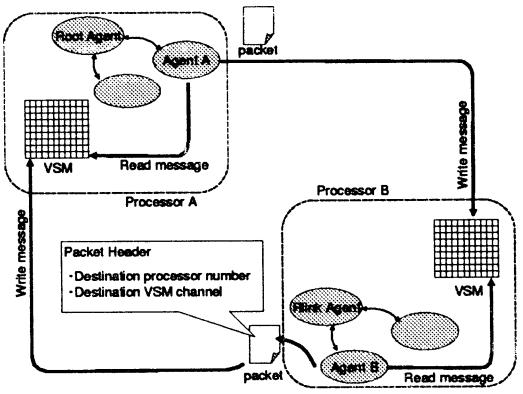


Figure 2: Real-time communication between arbitrary agents via network

with each other attempt to access the assigned channel to establish communication between them. For example, Linda achieves communication and synchronization of processes in a distributed computing environment using virtual shared memory (VSM).

However, in the RT-MRL execution system, we design a communication mechanism to prioritize real-time communication by describing the details of the shared memory for agent communication as follows Shared-memory spaces are allocated in all computers Agent communication is realized by each agent writing a message into another shared memory. Figure illustrates agent communication executed in different computers via the real-time network Responsive Link Responsive Link has a packet-routing function in hardware. A packet's routing is determined by comparing its header and contents of a routing table. The routing table consists of information packets composed of the source address, destination address, priority and a port number from which a packet is output. Packet routing and prioritization are realized in hardware The RT-MRL execution system determines if a network interface is used by referring to the destination address of a packet. In distributed agent communication, the RT-MRL execution system creates a packet automatically using information of priority, destination, and source address of a message. Using the above process, an RT-MRL developer can describe the agent communication behavior without a distributed computing environment.

4 Experiment and Discussion

In this section, we present the experiment and its results. We show the required time for establishing P2P communication between arbitrary agents as an experiment to evaluate our system. A dynamic interaction between agents, such as negotiations, is initiated by establishing P2P connection. It is difficult that the realization of hard real-time agent negotiation, while

the performance of hard real-time in the execution of agents and agent communications is realized using RTOS and real-time network. However, in order to prove that RT-MRL has sufficient performance to support a real robot system, we should demonstrate the time required for agents to interact during P2P processing. The flow of the P2P processing is initiated by broadcasting a message that means that arbitrary agent A is requesting an agent network for connection with specified agent B. Through the broadcast, when agent B receives the message from agent A, agent A sends a message meaning it accepts connection to agent B. Dynamic P2P connection between agents is then established through the RT-MRL function that enables allocating a channel in the shared memory.

The experiment result (Fig. 2) shows the average time for P2P connection establishment. Here, we use the Manhattan distance as an evaluation parameter. The connection form of the Responsive Link is one-to-one. A packet is thus transmitted to a destination computer through several processors. Therefore, when a packet moves to a neighboring node, Manhattan distance D is equal to 1. In the graphs, the y-axis represents the number of agents in our multiagent system. Graphs in Fig. 2 are the results for Manhattan distances of 1, 2, and 4. Two agents perform P2P communication in a 10msec period. As the number of agents increases, the required time for establishing P2P connection also increases slightly. This is caused by the increased time for a request message to reach a specific agent due to the increased number of agents. However, the required time is adequate for practical reactions in the real world. Moreover, these graphs demonstrate that the time required for P2P connection is independent of the Manhattan distance.

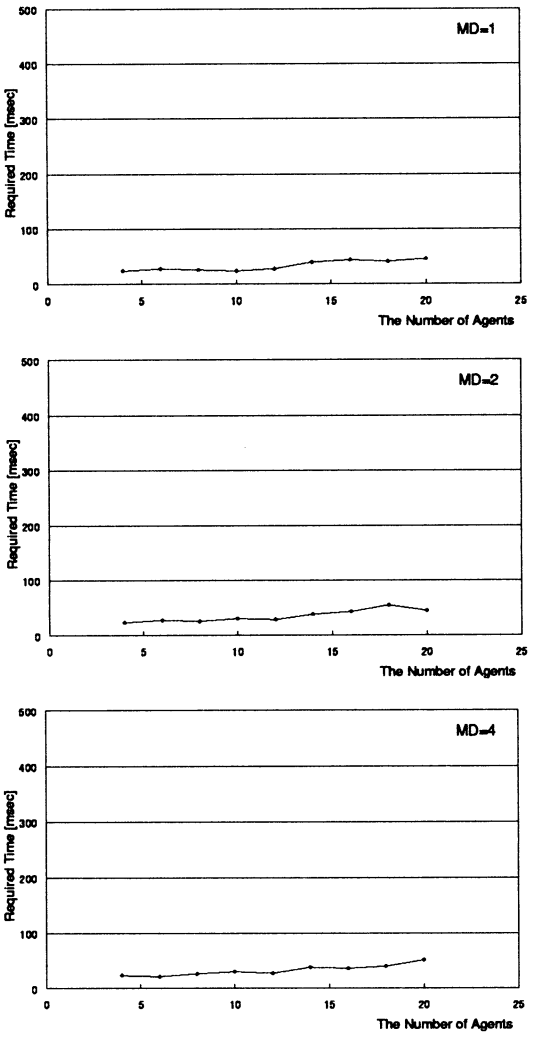


Figure 3: The required time for an establishment of P2P communicaion (Manhattan distance : Upper4, Middle2, Bottom1)

5 Conclusion

In this paper, we developed a multiagent system (RT-MRL) that enables constructing a robot system. RT-MRL can create an agent process that adds autonomy to processes. It can also integrate an asynchronous agent that controls various hardware or performs computational tasks. Moreover, all agents in an RT-MRL system can perform in hard real-time and realize hard real-time communication using a real-time network interface. An RT-MRL system can also be executed in an embedded computer, such as for small robots and information appliances. As a result, an RT-MRL system can realize integration of components in a distributed system, dynamic communication of distributed software, real-time execution of all processes, and inter-process communication. Experiment results demonstrated that such an RT-MRL system is suitable for application in a real robot system.

References

[1] Hiroyuki Nishiyama, Wataru Yamazaki,and Fumio Mizoguchi, "Multi-agent Language (MRL) based on Concurrent Logic Programming,", Computer Software, vol.20, No.1, pp.36-50, 2003, 1.

[2] Nobuyuki Yamasaki, Toshihiro Matsui, "Responsive Processor for Parallel/Distributed Real-Time Control," The Journal of the Robotics Society of Japan, vol.19, No.3, pp.68-77, 2001.

[3] Yousuke Matsusaka, Kentaro Oku, and Tetsunori Kobayashi "Design and Implementation of Data Sharing Architecture for Multi-Functional Robot Development," The Transactions of The Institute of Electronics, Information and Communication Engineers D-I Vol.J86-D-I No.5, pp.318-329, 2003.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

State Observer - Based Robust Control Scheme for Wheeled Mobile Robots

Masahiro Oya, Kenji Higashi, Makoto Wada, and Toshihiro Kobayashi
Department of Control Engineering, Kyushu Institute of Technology, Tobata, Kitakyushu, 804-8550, Japan E-mail: b344217k@tobata.isc.kyutech.ac.jp

Abstract

If the driving matrix B is positive definite matrix in n-input n-output systems such as $\dot{x}(t)=Ax(t)+Bu(t)$ $\in R^n$, it is very easy to design robust controller for the systems even if the driving matrix B is unknown. Using the fact, a new robust trajectory-tracking controller for wheeled mobile robots without velocity measurement is developed. It is shown theoretically that the proposed controller is robust not only for the weight and the moment of inertia but also for radius of wheels and distance from the reference point to wheels.

1. Introduction

Trajectory tracking control of wheeled mobile robot has received considerable attention and several schemes have been developed. In early works, schemes based on kinematics model have been proposed [1-2]. Recently, robust controllers taking into account dynamics such as weight and moment of inertia have been developed [3-10]. These controllers require full state measurements. In general, full state measurements may not be available due to cost of sensors, weight limitation, effects of noises, etc. Most recently, control schemes without using velocity measurements were proposed[11]. However, the exact information of radius wheels and distance from a reference point to both wheels are required in the schemes. Therefore, in the case when radius of rubber tire change due to the changes of weight and the position of center of gravity or there exist measurement errors in the distance from a reference point to wheels, the proposed schemes in [11], [12] can't be applied.

In this paper, we will propose a new robust tracking control scheme with the use of a state observer. The developed controller has the following features; i) The proposed controller has robustness not only for the weight and the moment of inertia but also for radiuses of wheels and distance from the reference point to wheels. ii) It is theoretically guaranteed that tracking performance can be easily improved by using design parameters.

2. Controlled object and trajectory tracking error equation

2.1 Controlled object

Wheeled mobile robot considered here is shown in **Fig.**1. Point P is the reference point, and point C is the center of gravity. Symbols $r_i\,(i=1,2)$ are radiuses of left and right wheel, symbols $\ell_i\,(i=1,2)$ denote distances from the reference point P to each wheel, and symbols a, d denote the distance from the reference point P to the center of gravity C. v_p, v_c are the velocities at each point in the direction of mobile robot axis L, and θ is the angle between mobile robot axis L and the axis X. It is assumed that wheeled mobile robot does not slide in the direction of wheel axis W. Then, considering the relation $v_c = v_p - a\dot{\theta}$, the dynamic equation of a wheeled mobile robot can be obtained as

$$\dot{q} = S(q)v, \quad M_p\dot{v} + C_p(\dot{q})v = B_p\tau, \tag{1}$$

$$\left.\begin{aligned} &M_p = (H^T)^{-1}MH^{-1}, C_p(\dot{q}) = \begin{bmatrix} 0 & -md\dot{\theta} \\ md\dot{\theta} & 0 \end{bmatrix} \\ &B_p = \Gamma R, M = \mathrm{diag}[m, I_c + md^2], R = \mathrm{diag}[1/r_1, 1/r_2] \\ &S(q) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}, \Gamma = \begin{bmatrix} 1 & 1 \\ \ell_1 & -\ell_2 \end{bmatrix}, H = \begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix} \\ &q^T = [x_p \;\; y_p \;\; \theta], v^T = [v_p \;\; d\theta/dt], \tau = [\tau_R \;\; \tau_L]^T \end{aligned}\right\}, \tag{2}$$

where x_p, y_p are the position of the reference point P along the X and Y Cartesian coordinate frames, m, I_c are mass of the mobile robot and the moment of inertia. τ is the torque input vector added to wheels.

2.2 Trajectory tracking error equation

The objective considered here is to make the reference point P and the angle θ track a desired trajectory without using velocity measurement v. To develop controller achieving the objective, tracking error equation is derived below.

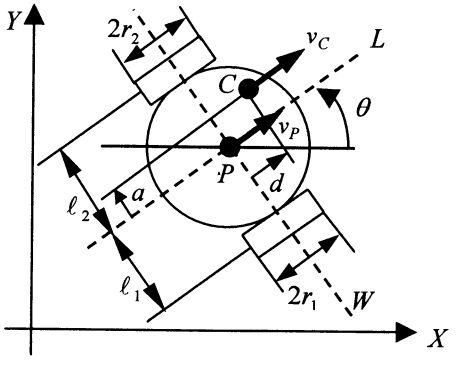


Fig.1 A wheeled mobile robot

It is assumed that desired trajectories are given by the form of $\dot{q}_d = S(q_d)v_d$, $q_d = [x_d \;\; y_d \;\; \theta]^T$ and $v_d = [v_{d1} \;\; v_{d2}]^T$. Tracking errors are defined by $\tilde{x} = x_p - x_d$, $\tilde{y} = y_p - y_d$ and $\tilde{\theta} = \theta - \theta_d$. Then, to make development of controller easy, the following variables $[\omega \;\; z^T]^T \in R^3$, $z = [z_1 \;\; z_2]^T \in R^2$ and u are defined [11].

$$\begin{bmatrix} \omega \\ z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} -\tilde{\theta}\cos\theta + 2\sin\theta & -\tilde{\theta}\sin\theta - 2\cos\theta & 0 \\ 0 & 0 & 1 \\ \cos\theta & \sin\theta & 0 \end{bmatrix} \tilde{q}, \quad (3)$$

$$u = T^{-1}v - \begin{bmatrix} v_{d2} \\ v_{d1}\cos z_1 \end{bmatrix}, T = \begin{bmatrix} 0.5(\omega + z_1 z_2) & 1 \\ 1 & 0 \end{bmatrix}. \quad (4)$$

After taking the time derivative (3), the tracking error system can be obtained as

$$\left.\begin{aligned} &\dot{\omega} = u^T J^T z + f \;,\; \dot{z} = u \\ &J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},\; f = 2\left(v_{d2} z_2 - v_{d1}\sin z_1\right) \end{aligned}\right\}. \quad (5)$$

It is supposed that u can be treated as a control input, the input u that can actually stabilize the closed loop system is given by (see [11] of chapter 2)

$$u = u_a - k_2 z, \quad (6)$$

$$\left.\begin{aligned} &u_a = \sigma J z_d + \Omega_1 z_d,\; \sigma = \frac{k_1\omega + f}{\delta_d^2} \\ &\Omega_1 = k_2 + \frac{\dot{\delta}_d}{\delta_d} + \omega\sigma,\; \delta_d = \alpha_0 e^{(-\alpha_1 t)} + \varepsilon_1 \\ &\dot{z}_d = \frac{\dot{\delta}_d}{\delta_d} z_d + (\sigma + \omega\Omega_1) J z_d,\; \| z_d(0) \|^2 = \delta_d(0)^2 \end{aligned}\right\}, \quad (7)$$

where $k_i \geq 0, \alpha_i \geq 0$ and $\varepsilon_1 > 0$ are design parameters. The parameter ε_1 is introduced to assure boundedness of the signal σ in the case when the tracking error ω and z don't converge to zero. As shown in [11], it is assured that tracking errors ω and z exponentially converge to a neighborhood around zero that can be made arbitrarily small.

It is obvious from (1) and (4) that the signal u can't be used as a control input signal. Therefore, let the signal (6) be ideal signal u_d, and then, let's develop an input torque to make the signal u track the ideal signal u_d. To this end, the error signals $\eta = [\eta_1 \;\; \eta_2]^T = u_d - u, \tilde{z} = [\tilde{z}_1 \;\; \tilde{z}_2]^T = z_d - z$ are defined. Using the same analysis stated in [11](chapter 1.6), the error system is obtained as

$$\dot{\omega} = -k_1\omega + u_a^T J\tilde{z} + \eta^T J z, \quad (8)$$

$$\dot{\tilde{z}} = -k_2\tilde{z} + \omega J u_a + \dot{\eta}, \quad (9)$$

$$\overline{M}\dot{\eta} = -\overline{V}_m\eta - \overline{V}_f\eta - T^T\Gamma R\tau + \mu_1, \quad (10)$$

$$\left.\begin{aligned} &\overline{M} = T^T M_p T,\; \overline{V}_m = T^T M_p \dot{T},\; \overline{V}_f = T^T C_p T \\ &\mu_1 = \overline{M}\,\dot{u}_d + \overline{V}_m u_d + \overline{V}_f u_d + T^T M_p \dot{\Pi} + T^T C_p \Pi \\ &\Pi = \begin{bmatrix} v_{d1}\cos z_1 + 0.5 v_{d2}(\omega + z_1 z_2) \\ v_{d2} \end{bmatrix} \end{aligned}\right\} \quad (11)$$

In the following, based on the error system (8)-(11), a controller is developed achieving the design objective. There exist the following characteristics P1 and P2 exist for the error system (8)-(11).

[P1] The matrix M_p is symmetric positive definite and there exist bounded constant values $\underline{\rho}_{mp}$ and $\overline{\rho}_{mp}$ such that

$$\underline{\rho}_{mp} x^T x \leq x^T M_p x \leq \overline{\rho}_{mp} x^T x, \text{ for } \forall\; x. \quad (12)$$

[P2] $\overline{M}$ is a positive definite matrix and the following relation is satisfied as

$$x^T \left(d\overline{M}/dt - 2(\overline{V}_m + \overline{V}_f)\right)x = 0. \quad (13)$$

3. Construction of Controller

If norms of the tracking error ω and $\tilde{z}$ become small, the norm of the tracking error $\tilde{q}$ becomes also small. Therefore, if a controller is developed so that norms of the tracking error ω and $\tilde{z}$ become small, the design objective can be achieved. To meet this, a following controller is proposed.

$$\tau = k_\eta \hat{\Gamma}^T T\hat{\eta}, \quad (14)$$

$$\left.\begin{aligned} &\hat{\eta} = -k_{\tilde{\eta}} z + u_d + \xi,\; \xi(0) = k_{\tilde{\eta}} z(0) \\ &\dot{\xi} = -k_{\tilde{\eta}}\left(-u_d + \hat{\eta}\right) - T^{-1}\hat{T}_1\left(\hat{\eta} - u_d\right) \end{aligned}\right\}, \quad (15)$$

where $\hat{\eta}$ is the estimated signal of η , k_η is the feedback gain, $k_{\tilde{\eta}}$ is the observer gain and $\hat{T}_1, \hat{\Gamma}$ are estimated matrices for $T_1 (= \dot{T})$ and Γ , respectively.

$$\left.\begin{aligned} &\hat{T}_1 = \dot{T} - \begin{bmatrix} 0.5\left(\eta^T J z - \eta_1 z_2 - \eta_2 z_1\right) & 0 \\ 0 & 0 \end{bmatrix} \\ &\hat{\Gamma} = \begin{bmatrix} 1 & 1 \\ \hat{\ell}_1 & -\hat{\ell}_2 \end{bmatrix} \end{aligned}\right\} \quad (16)$$

Estimated error are defined by $\tilde{\eta} = \eta - \hat{\eta}$ and $\tilde{V}_m = \overline{V}_m - T^T M_p \hat{T}_1$. Then, estimated error equation is given by

$$\left.\begin{aligned} &\overline{M}\dot{\tilde{\eta}} = -k_{\tilde{\eta}}\overline{M}\tilde{\eta} - \tilde{V}_m\tilde{\eta} - k_\eta T^T\Gamma R\hat{\Gamma}^T T\hat{\eta} + \mu_2 \\ &\mu_2 = -\tilde{V}_m(\eta - \tilde{\eta} - u_d) - \overline{V}_f(\eta - u_d) \\ &\qquad + T^T M_p \dot{\Pi} + T^T C_p \Pi \end{aligned}\right\}. \quad (17)$$

As shown in the theorem below, the design objective can be accomplished using the controller (14)-(15). To show this, the characteristics are added and some assumptions are made for the error system (8)-(11). Furthermore, the monotone increasing function used in the theorem is explained.

[P3] Using the definition $\tilde{\Gamma} = \Gamma - \hat{\Gamma}$, $\Gamma R\hat{\Gamma}^T$ can be expressed as $\Gamma R\hat{\Gamma}^T = B_1 - B_2 + B_3$, $B_1 = 0.25(\Gamma + \hat{\Gamma})R(\Gamma + \hat{\Gamma})^T$, $B_2 = 0.25\tilde{\Gamma}R\tilde{\Gamma}^T$, $B_3 = 0.25\tilde{\Gamma}R(\Gamma + \hat{\Gamma})^T - 0.25(\Gamma + \hat{\Gamma})R\tilde{\Gamma}^T$. Where, B_1 is a positive definite matrix, B_2 is a semi-positive definite matrix, and B_3 is a skew symmetric matrix.

[P4] The relation $\|z_d\|^2 = \delta_d^2 \leq \overline{\rho}_z^2 = \|z_d(0)\|^2$ is satisfied.

[A1] There exist known bounded constant values $\overline{\rho}_{i,j}\,(i = 1,2,\; j = 1,\cdots,9)$ satisfying the relation

$$\left.\begin{aligned}&\|\mu_i\| \le (k_1+k_2+1)^2 \sum_{j=0}^{9} \bar{\rho}_{i,j}\|e\|^j, i=1,2\\ &e^T = [\omega, \tilde{z}, \eta^T T^T M_p^{1/2}/(k_1+k_2+1), \tilde{\eta}^T T^T M_p^{1/2}/(k_1+k_2+1)]\end{aligned}\right\}. \quad (18)$$

[A2] The matrix $B_1 - 4B_2$ is a positive definite, and there exist known bounded values $\underline{\rho}_{b1}$ and $\bar{\rho}_{b1}$ satisfying the relation

$$\underline{\rho}_{b1}\|x\|^2 \le x^T(B_1 - 4B_2)x \le \bar{\rho}_{b1}\|x\|^2, \text{ for } \forall x. \quad (19)$$

[A3] There exist a known bounded constant value $\bar{\rho}_{b2}$ such that $\|2B_1 + B_2\| \le \bar{\rho}_{b2}$.

[A4] Positive constants $\underline{\rho}_{mp}$ and $\bar{\rho}_{mp}$ are known.

[A5] There exists a known bounded constant value $\bar{\rho}_{v0}$ such that $V(0) \le \bar{\rho}_{v0}$, where $V = e^T e$.

In assumption A1, it is used that there exists a known bounded constant value $\bar{\rho}_T$ such that $\|T^{-1}\| \le \bar{\rho}_T \sum_{j=0}^{2}\|e\|^j$ and the relation $0 < \varepsilon_1 \le \delta_d \le \alpha_0 + \varepsilon_1$ holds. It should be noted that the values of $\bar{\rho}_{i,j}$ are determined regardless of the values of the design parameters $k_i, k_\eta, k_{\tilde{\eta}}$ and α_1. In assumption A2, it is noted that the relation $B_1 - 4B_2 > 0$ holds if the sufficient condition $8(\tilde{\ell}_1^2 + \tilde{\ell}_2^2)/(\ell_1 + \hat{\ell}_1 + \ell_2 + \hat{\ell}_2)^2 < \lambda_{\min}[R]/\lambda_{\max}[R]$ is satisfied where $\tilde{\ell}_i$ is the estimated error given by $\tilde{\ell}_i = \ell_i - \hat{\ell}_i$. In assumption A5, it is used that there exists a known positive constant $\bar{\rho}_{ud}$ such that $\|u_d(0)\| \le (k_1 + k_2 + 1)\bar{\rho}_{ud}$. From this it is obvious that there exists a known bounded constant value $\bar{\rho}_{v0}$ independent of the design parameters $k_i, k_\eta, k_{\tilde{\eta}}$ and α_1.

The monotonically increasing function $g(V)$ is defined as

$$\left.\begin{aligned}g(V) &= (1+\bar{\rho}_z^2)\bar{\rho}_T^2 \sum_{j=1}^{4} V^j + \bar{\rho}_T^2 \sum_{j=1}^{3} V^j + \bar{\rho}_T^2 (g_1(V) + g_2(V))\\ g_i(V) &= \bar{\rho}_{i,0}^2 + (\bar{\rho}_{i,0} + \bar{\rho}_{i,1})^2 V + \sum_{j=2}^{9} (\bar{\rho}_{i,j} + \bar{\rho}_{i,j-1} + \bar{\rho}_{i,j-2})^2 V^j\\ &\quad + (\bar{\rho}_{i,8} + \bar{\rho}_{i,9})^2 V^{10} + \bar{\rho}_{i,9} V^{11}\end{aligned}\right\}. \quad (20)$$

As shown in **Fig.2**, there obviously exists lower bound of the design parameter $\beta = \underline{\beta} \ge 0$ such that the equation $-\beta V + g(V) = 0$ has two different positive real roots $V_i, i = 1,2$. Then, in the range $V \in [V_1, V_2]$, the inequality $-\beta V + g(V) \le 0$ holds. It should be noted that if the design parameter β is larger than $\underline{\beta}$, V_1 becomes small and V_2 becomes larger.

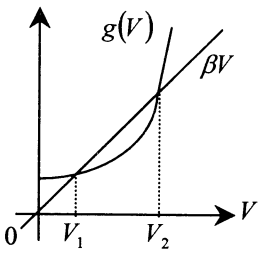


Fig.2 Real roots of equation $-\beta V + g(V) = 0$

<Theorem 1> The design parameters are set to

$$\left.\begin{aligned}&k_1 = k_2 = 0.5\beta, k_\eta = \frac{\bar{\rho}_{mp}\beta + 18(k_1+k_2+1)^2}{\underline{\rho}_{b1}}\\ &k_{\tilde{\eta}} = \frac{1}{2}\beta + \frac{\bar{\rho}_{b2}k_\eta + 6(k_1+k_2+1)^2}{\underline{\rho}_{mp}}\end{aligned}\right\}, \quad (21)$$

and the design parameters except β and α_1 are set to some fixed values. It is assumed that $V(0) \le \bar{\rho}_{v0} < V_2$ holds where V_2 denotes the positive real root (see **Fig.2**) of the equation $-\beta V + g(V) = 0$ with respect to some fixed parameter $\beta = \underline{\beta} > 0$. Then, for any design parameters $\beta \ge \underline{\beta}$ and $\alpha_1 \ge 0$, the closed loop system using the controller (14), (15) becomes stable, and the following relation is satisfied.

$$\left.\begin{aligned}&\|z\| \le \sqrt{\bar{\rho}_{v0}e^{-\beta t} + \frac{g(\bar{\rho}_g)}{\beta}} + \alpha_0 e^{-\alpha_1 t} + \varepsilon_1\\ &|\omega| \le \sqrt{\bar{\rho}_{v0}e^{-\beta t} + \frac{g(\bar{\rho}_g)}{\beta}}, \bar{\rho}_g = \max[\bar{\rho}_{v0}, V_1]\end{aligned}\right\} \quad (22)$$

△△

It follows from Theorem 1 that the ultimate value of norm $\|z\|$ is $\|z\| \le \sqrt{g(\bar{\rho}_g)/\beta} + \varepsilon_1$. From this it is concluded that $\|z\|$ converges to any small ultimate value with any convergent rate by setting design parameters β and ε_1.

(Proof) Using P1-P4, A1-A4 and the following relations

$$\left.\begin{aligned}&4\eta^T T^T B_3 T\tilde{\eta} \le 4\tilde{\eta}^T T^T B_2 T\tilde{\eta} + \eta^T T^T B_1 T\eta\\ &\qquad + 2\tilde{\eta}^T T^T B_1 T\tilde{\eta} + 2\eta^T T^T B_2 T\eta\\ &2\omega\eta^T J z \le \|T\eta\|(\|e\|^2 + 2\bar{\rho}_z\|e\|)\|T^{-1}\|\\ &\qquad \le 3\|T\eta\|^2 + \bar{\rho}_T^2(1+2\bar{\rho}_z)^2\sum_{j=1}^{4} V^j\\ &2\tilde{z}^T\eta \le 3\|T\eta\|^2 + \bar{\rho}_T^2\sum_{j=1}^{3} V^j\\ &2\eta^T\mu_1(k_1+k_2+1)^{-2} \le 12\|T\eta\|^2 + \bar{\rho}_T^2 g_1(V)\\ &2\tilde{\eta}^T\mu_2(k_1+k_2+1)^{-2} \le 12\|T\tilde{\eta}\|^2 + \bar{\rho}_T^2 g_2(V)\end{aligned}\right\}, \quad (23)$$

it can be seen the time derivative of $V = e^T e$ satisfies

$$\dot{V} \le -\beta V + g(V). \quad (24)$$

First, let's consider the case of $V(0) \le V_1$. The right-hand side of an equation (24) doesn't become negative at the initial time $t = 0$. However, since the right-hand side of an equation (24) becomes negative in the range $V \in [V_1, V_2]$, V remains in the domain of $V \le V_1$. Next, let's consider the case of $V_1 \le V(0) \le V_2$. Since the right-hand side of an equation (24) becomes negative in the range $V \in [V_1, V_2]$, V decreases in the range $V \in [V_1, V_2]$. Even if $V(t)$ satisfies $V(t_1) \le V_1$ at a certain time $t = t_1$, V remains in the domain of $V(t) \le V_1$ after the time $t = t_1$. It can be concluded from the fact stated above and the relation $V(0) \le \bar{\rho}_{v0} < V_2$ that the closed loop system becomes stable and the relation $V \le \bar{\rho}_g$ is satisfied. Finally, analyzing the derivative of V again, the relation $\dot{V} \le -\beta V + g(\bar{\rho}_g)$ can be obtained. From the relation, the inequality (22) can be derived.

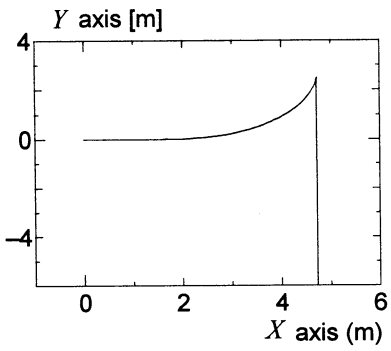


Fig.3 Reference trajectory

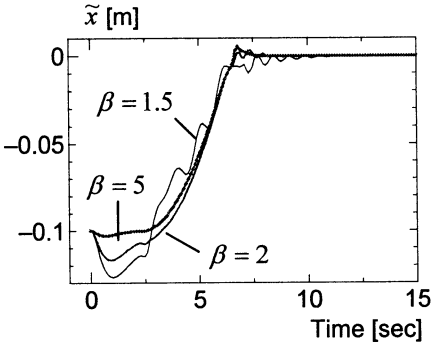


Fig.4 Position tracking error $\tilde{x}$

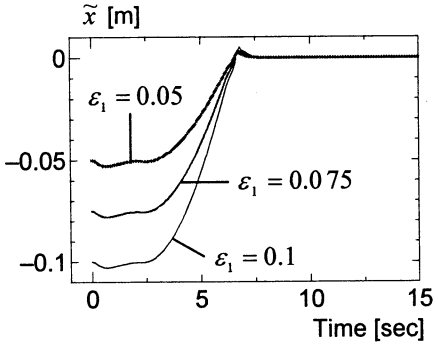


Fig.5 Comparison in the case of $\varepsilon_1 = 0.1, 0.075, 0.05$

4. Simulation Results

In this simulation, the controller (14)-(15) is designed for a wheeled mobile robot shown in **Fig.1**. The values of system parameters of the mobile robot are given by mass of robot: $m = 30[\mathrm{kg}]$, moment of the inertia around axis passing through point $C: I_c = 10[\mathrm{kgm}^2]$, radius of left and right wheels: $r_1 = 0.12[\mathrm{m}], r_2 = 0.08[\mathrm{m}]$, distance from the reference point to left and right wheels: $\ell_1 = 0.3[\mathrm{m}], \ell_2 = 0.2[\mathrm{m}]$ and distance from the reference point to the center of gravity: $a = 0[\mathrm{m}]$, $d = 0.2[\mathrm{m}]$. The design parameters are selected as $\alpha_0 = \alpha_1 = 0$. The estimates for parameters $m, I_c, r_1, r_2 \ell_1, \ell_2, a$ and d of the mobile robot are $\hat{m} = 20[\mathrm{kg}]$, $\hat{I}_c = 7.0[\mathrm{kgm}^2]$, $\hat{r}_1 = \hat{r}_2 = 0.1[\mathrm{m}]$, $\hat{\ell}_1 = \hat{\ell}_2 = 0.25[\mathrm{m}]$ and $a = d = 0[\mathrm{m}]$. **Fig.3** shows the reference trajectory in $X-Y$ plane. The initial value of auxiliary signal $z_d(t)$ is $z_d(0) = [0 \quad -\varepsilon_1]^T$.

Fig.4 shows trajectory tracking error responses for $\beta = 1.5, 2, 5$ and $\varepsilon_1 = 0.1$. **Fig.5** shows trajectory tracking error responses for $\varepsilon_1 = 0.1, 0.05, 0.01$ and $\beta = 5$. As shown in **Figs.4** and **5**, it is concluded that the tracking performance can be easily improved by using design parameter β and ε_1. Also for $\tilde{y}$ and $\tilde{\theta}$, tracking performance can be improved by using design parameters β and ε_1.

5. Conclusion

In this paper, we proposed the robust trajectory tracking control schemes using only the position information for wheeled mobile robot. In theoretical analysis, it have been shown that the closed loop system has robustness not only for the weight and the moment of inertia but also for radius of wheels and distance from the reference point to wheels. It is also shown that tracking performance can be easily improved in the closed loop system by using the design parameters β and ε_1.

Reference

[1] Z. P. Jiang, and H. Nijmeijer, Tracking Control of Mobile Robots: A Case Study in Backstepping, Automatica, Vol. 33, pp. 1393-1397, (1997)

[2] W. Wu, H. Chen, Y. Wang, and P. Woo, Adaptive Exponential Stabilization of Mobile Robots with Uncertainties, Proc. IEEE Conf. Decision and Control, pp. 3484-3489, (1999)

[3] R. Fierro, and F. L. Lewis, Control of A Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics, J. Robotic Systems, Vol. 14, pp. 149-163, (1997)

[4] R. Fierro, and F. L. Lewis, Control of A Nonholonomic Mobile Robot Using Neural Networks, IEEE Trans. On Neural Networks, Vol. 9, pp.589-600, (1998)

[5] T. C. Lee, C. H. Lee, and C. C. Teng, Adaptive Tracking Control of Nonholonomic Mobile Robots by Computed Torque, Proc. IEEE Conf. Decision and Control, pp. 1254-1259, (1999)

[6] S. Lin, and A. Goldenberg, Robust Damping Control of Wheeled Mobile Robots, Proc. IEEE Conf. Robotics and Automation, pp. 2919-2924, (2000)

[7] W. E. Dixon, D. M. Dawson, F. Zhang, and E. Zergeroglu, Global Exponential Tracking Control of a Mobile Robot System via a PE condition, IEEE Trans. on Systems, Man and Cybernetics, Vol.30, pp. 129-142, (2000)

[8] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and F. Zhang, Robust Tracking and Regulation Control for Mobile Robots, Int. J. Robust and Nonlinear Control,Vol. 10, pp. 199-216, (2000)

[9] W.E.Dixon, M.S.de Queiroz, and D. M. Dawson, Adaptive Tracking and Regulation of a Wheeled Mobile Robot with Controller /Update Law Modularity, Proc.IEEE Conf. Robotics and Automation, pp. 2620-2625, (2002)

[10] D. K. Chwa, J. H. Seo, P. Kim, and J. Y. Choi, Sliding Mode Tracking Control of Nonholonomic Wheeled Mobile Robots, Proceedings of the ACC. pp.3991-3996, (2002)

[11] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and A. Behal, Nonlinear Control of Wheeled Mobile Robots, Lecture Notes in Control and Information Science 262, Springer, (2001)

[12] A. P. Aguiar, A. N. Atassi, and A. M. Pascoal, Regulation of a Nonholonomic Dynamic Wheeled Mobile Robot with Parametric Modeling Uncertainty using Lyapunov Function, Proc. IEEE Conf. Decision and Control, pp. 2995-3000, (2000)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Estimation of vehicle cornering stiffness of using GPS/INS

Munki Lee*, Gunhong Park**, Yushin Chang**, Sinpyo Hong***, and Manhyung, Lee***

*Department of Interdisciplinary Program in Mechatronics, Pusan University, Pusan, Korea
**Department of Mechanical and Intelligent Systems Engineering., Pusan University, Pusan, Korea
***School of Mechanical Engineering, Pusan University, Pusan, Korea

E-mail: ghpak@pusan.ac.kr ,mahlee@pusan.ac.kr

Abstract: This paper demonstrates a unique method for measuring vehicle states such as body sideslip angle and tire sideslip angle using Global Positioning System (GPS) velocity information in conjunction with other sensors. A method for integrating Inertial Navigation System (INS) sensors with GPS measurements to provide higher update rate estimates of the vehicle states is presented, and the method can be used to estimate the tire cornering stiffness. The experimental results for the GPS velocity-based sideslip angle measurement.

Keywords: GPS, INS, Cornering stiffness, Side slip angle, Vehicle, parameter estimation

1. INTRODUCTION

To design estimator of vehicle and road state, it is necessary to estimate sideslip angle and cornering stiffness and essential physical parameter to make VDC system [1]. Sideslip angle can be estimate to use yaw angular velocity estimator[2], non-linear estimator and inertial measurement sensor[3][4]. But when vehicle approaches to neutral steering that vehicle maintain constant circular rotation and front wheel and real wheel of vehicle is taken lateral force, it is impossible to measure sideslip angle from yaw angular velocity and steering angle. It is not easy to measure by direct sensor measurement directly. When the vehicle cruises at high speed, the tire also rotates at high speed. It is hard to mount a precision sensor to vehicle. To solve this problem, there are methods to estimate important variables indirectly. Masmoudi and Hedrick estimated the axis of vehicle torque by using sliding mode estimator[5], Ray estimated tire force by using Kalman filter and using that force estimated the road friction coefficient[6]. And observer is used to estimate state variable. Huh and Stein define the gain to get well-condition by scaling the system matrix and designs the observer that robust to noise[7]. But these methods have many restrict conditions so it is hard to apply to real situation.

Therefore, in this paper, we use the new method to measure the sideslip and cornering stiffness. From GPS/INS navigation system we can get the position and velocity of vehicle. A method for integrating sensors with GPS measurements to provide higher update rate estimates of the vehicle states is presented can be used to estimate the cornering stiffness of tire. The experimental results for the GPS/INS velocity-based sideslip angle measurement.

2. STATE MEASUREMENT

The bicycle model shown in Figure 1 is 2 DOF vehicle model in a plane with lateral velocity and yaw rate as the states. The lateral and yaw motions are described in Equation (1)

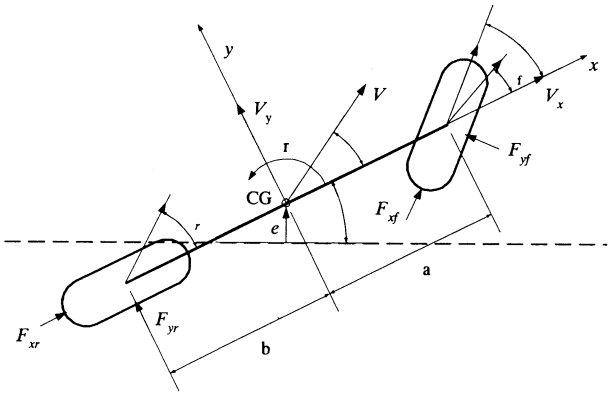


Fig. 1 Simple Bicycle Model of Vehicle

$$\begin{bmatrix} \dot{V}_y \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{-C_{\alpha f}-C_{\alpha r}}{mV_x} & -V_x+\left(\frac{C_{\alpha r}b-C_{\alpha f}a}{mV_x}\right) \\ \frac{C_{\alpha r}b-C_{\alpha f}a}{I_zV_x} & \frac{-C_{\alpha f}a^2-C_{\alpha r}b^2}{I_zV_x} \end{bmatrix} \begin{bmatrix} V_y \\ r \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha f}}{m} \\ \frac{C_{\alpha f}a}{I_z} \end{bmatrix} \delta \quad (1)$$

x, y: frame that based from C.G of vehicle
V : velocity vector of vehicle
F_{xf}, F_{xr} : force that act to front and rear wheel to tire's latitude axis
F_{yf}, F_{yr} : force that act to front and rear wheel to tire's longitudinal axis
r : yaw rate of vehicle
a, b : distance from front and rear wheel to C.G of vehicle
a_f, a_r : side slip angel of front and rear wheel
b : angle between center line and velocity vector from C.G. of vehicle
d : steering angle of front wheel
y : angle between Earth-centered earth-fixed frame

and vehicle frame.

Using that, lateral slip of vehicle can be measured.

$$\beta = \psi_{GPS}^{VEL} - \psi \tag{2}$$

ψ_{GPS}^{VEL} is the angle of velocity vector of vehicle that measured by GPS. Angle between vehicle center line and earth frame is estimated by using yaw gyro. Gyro is initialized when vehicle go a straight line and bias can be estimated and removed. But bias is varied as time. To correct bias, attitude data of GPS is used, and corrects the ψ. Lateral velocity of vehicle can be calculated by following equation.

$$V_y^{GPS} = V^{GPS} \sin(\beta) \tag{3}$$

Slip angle that measured by GPS is slip angle of point that GPS antenna is mounted so it is not actual slip angle of C.G, front and rear wheel of vehicle. To get the velocity of point P by GPS antenna A, relative velocity vector from point A to point P is needed.

$$V_p = V_A + \vec{\omega} \times \vec{r}_{A/P} \tag{4}$$

ω includes roll, pitch and yaw of vehicle. we can get velocity of point P by using measurement of ω,. To get the slip angle of point P, we can use the following equation.

$$\beta_p = \tan^{-1}\left(\frac{V_Y^P}{V_X^P}\right) \tag{5}$$

Vx, Vy is velocity that measured from vehicle base coordinate. Slip angles of tire (α_f and α_r from Fig. 1) are the angle between actual tire direction and moving direction of tire. It is caused by difference of distance between C.G. axis of car. So using the velocity measurement of C.G. and distance from CG to car axis, velocity vector of front and rear tire can be estimated by using equation (4) and side slip angle of each tire can be estimated by using (5). In case of rear wheel, direction of tire and vehicle are same so side slip angle estimated above equation can be applied directly. But in case of front wheel, side slip angle contains steering angle so side slip angle obtains to remove steering angle.

$$\begin{aligned} \alpha_f &= \beta_f^{tire} - \delta \\ \alpha_r &= \beta_r^{tire} \end{aligned} \tag{6}$$

Cornering stiffness of front and rear wheel is estimated by using acceleration of GPS measurement through Kalman filtering. A lot of method exists to estimate

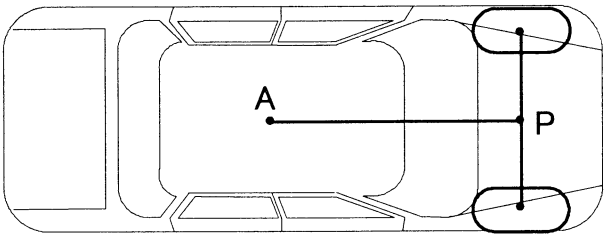


Fig. 2 Position of A and P

cornering stiffness. In this paper cornering stiffness can be obtain using following Newton equation in bicycle model.

$$\begin{aligned} \sum F_y &= m\ddot{y} = F_{yf} + F_{yf}\cos(\delta) \\ \sum M_z &= I_z\ddot{\psi} = aF_{yf} - bF_{yr}\cos(\delta) \end{aligned} \tag{7}$$

Force of side direction of front and rear tire is obtained by definition of cornering stiffness.

$$\begin{aligned} F_{yf} &= 2C_{\alpha f}\alpha_f \\ F_{yr} &= 2C_{\alpha r}\alpha_r \end{aligned} \tag{8}$$

Substitute (7) for (8), cornering stiffness can be obtain as following equations.

$$\begin{aligned} C_{\alpha f} &= \frac{bm\ddot{y} + I_z\ddot{\psi}}{2(a+b)\alpha_f} \\ C_{\alpha r} &= \frac{am\ddot{y} - I_z\ddot{\psi}}{2(a+b)\alpha_r\cos(\delta)} \end{aligned} \tag{9}$$

3.Error model for Kalman filter

Algorithm for integrate GPS/INS is followed[8]. State variable for system is defined as following

$$x = [\delta P^T \quad \delta V^T \quad \gamma^T \quad \varepsilon_g^T \quad \varepsilon_a^T]^T \tag{10}$$

Measurement is

$$y = \begin{bmatrix} \delta P_{GPS}^e \\ \delta\theta \end{bmatrix} \tag{11}$$

Then state equation for error model is

$$\begin{aligned} \dot{x} &= Ax + w \\ y &= Cx + v \end{aligned} \tag{12}$$

where,

$$A = \begin{bmatrix} 0 & I_3 & 0 & 0 & 0 \\ G & -\Omega_e & -RF & 0 & R \\ 0 & 0 & -\Omega & I_3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{13}$$

$$C = \begin{bmatrix} I_3 & 0 & -RL_1 & 0 & 0 \\ 0 & 0 & I_3 & 0 & 0 \end{bmatrix} \quad (14)$$

$$w = \begin{bmatrix} 0 & (Rw_a)^T & (w_g)^T & 0 & 0 \end{bmatrix}^T \quad (15)$$

$$v = [-v_p^T \quad -v_\theta^T]^T \quad (16)$$

Error of accelerometer and gyro is $0.05 m/s^2$ and $0.1 \times \pi / 180$ rad. Position and attitude error of GPS is 0.02m and $0.5 \times \pi / 180$ rad. Covariance matrix of process and measurement noise is as followed

$$E[ww^T] = R = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & (0.05)^2 I_3 & 0 & 0 & 0 \\ 0 & 0 & \left(0.1 \times \frac{\pi}{180}\right)^2 I_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (16)$$

$$E[vv^T] = Q = \begin{bmatrix} (0.02)^2 I_3 & 0 \\ 0 & \left(0.5 \times \frac{\pi}{180}\right)^2 I_3 \end{bmatrix} \quad (17)$$

4. Experiment

Cornering stiffness and side slip angle are measured by vehicle experiment. IMU, GPS receiver and PDL for RTK mounted to vehicle. IMU data and GPS data is received using serial interface. To get 3 difference serial data, serial expand card is used. Data of each sensor is recorded in Pentium PC. After recording, result of travel is acquired though post-processing. Post processing is used to get various travel data. This experiment takes place in Pusan National University playground. System consists of GPS/INS navigation system, steering angle measurement and PC that recording the whole data.. Experimental vehicle is KIA Sportage. Specification of this vehicle is as followed. Yaw moment and position of C.G. is used approximated value. .

Table 1. Specification of vehicle

Wheelbase(m)	2.65
a (distance from C.G. to front wheel, m)	1.4
b (distance from C.G. to rear wheel, m)	1.25
Curb weight (kg)	1465
Total yaw inertia($kg.m^2$)	2931

Two GPS are used. One is Z-Sensor of Ashtech. It is used to get the precise position. It has 1cm horizontal error and provides 1Hz measurement. The other is ADU2. It is used to get the attitude. It has 4 antennas and uses 12 channels. PDL of PACIFIC CREST is used to send and receive data between base and remote

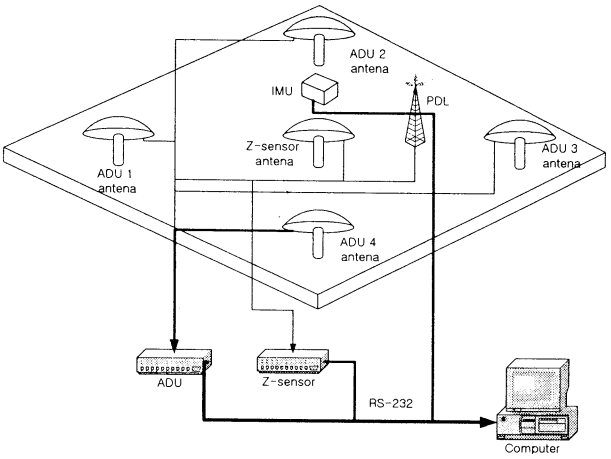


Fig. 3 Diagram of Measurement system at vehicle

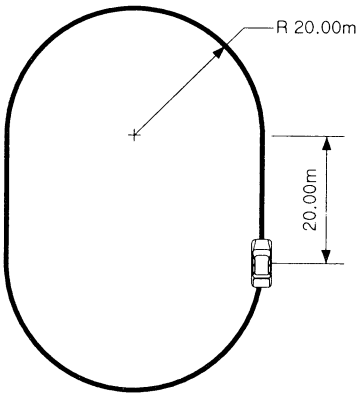


Fig. 4 Layout of test track

maximum transmitting speed is 19,200 Baud Rate. DMU-6X of Crossbow is used to get accelerations of 3-axis accelerometers and angular rate of 3 axis gryos. Maximum data transmitting rate is 38400 Baud Rate.
To measure steering angle of vehicle, potensiometer is equipped to steering wheel. Figure 3 shows diagram of measurement system at vehicle. Test vehicle follows the track that drawn in test field. Tracks shape is shown as follow Figure 5 shows the side slip angle that calculated by equation (5). Direction of traveling vehicle is counter clock wise direction so expected side slip angle direction is one side. But Figure 5 shows that side slip produces both sides. It observed when vehicle turns a corner and go through straight line. Speed of front wheel and rear wheel is calculated from equation (4) that substituted speed that measured center of gravity of vehicle. Side slip angle is estimated from substituting them to equation (5) and (6) Figure 6 shows side slip angle of front and rear wheel Front wheel is more side slipping than rear wheel. Cornering stiffness of front and rear wheel is acquired that substitute side slip angle to equation (9). Figure 7 shows the cornering stiffness of test vehicle. Mean cornering stiffness of front wheel and rear wheel is estimated 28300N/rad and 38400N/rad. General cornering stiffness on asphalt pavement is

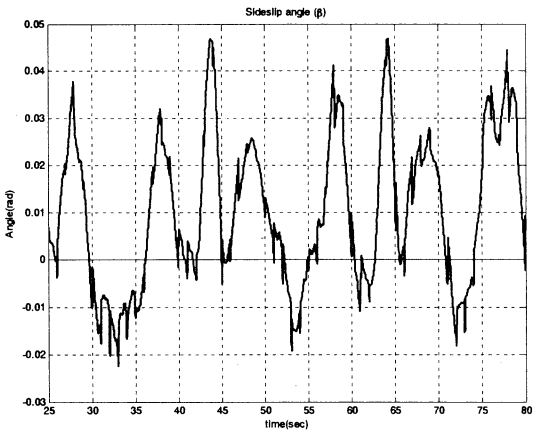


Fig. 5 Calculated sideslip angle

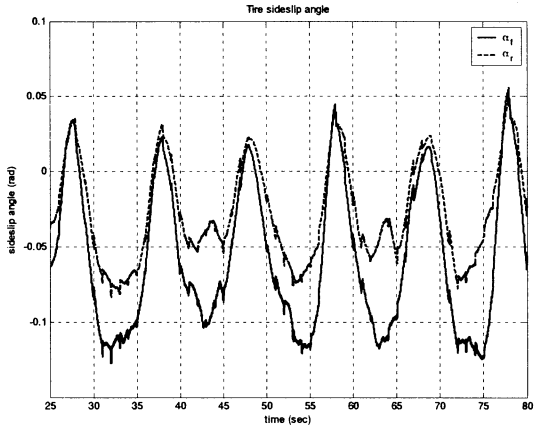


Fig. 6 Tire sideslip angle

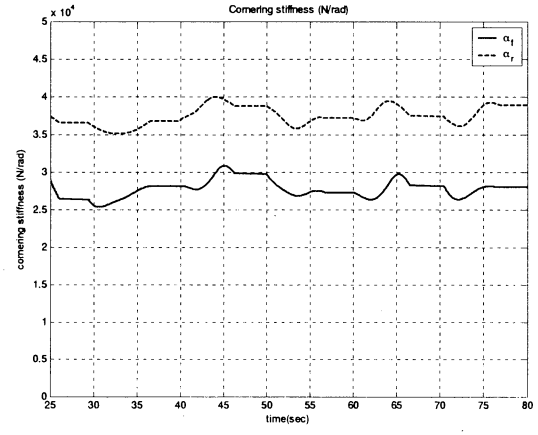


Fig. 7 Cornering stiffness of front and rear

60000N/rad. Because this experiment is on unpaved and wet road, Cornering stiffness of this experiment is lower than on an asphalt pavement. Cornering stiffness of front wheel is lower than rear wheel. Vehicle turn makes more side slipping for front wheel than rear wheel.

5. Conclusion

In this paper, it is shown that cornering stiffness that parameter of bicycle model can be estimated using GPS and INS integration. To integrate GPS and INS, Kalman filter is used. Using integrated GPS/INS system, position, velocity and attitude of vehicle is measured. Using these parameters, side slip angle and cornering stiffness is estimated. Side slip angle is estimated using yaw angle that measured gyro in IMU. It is more delicate than other estimation method. To estimate side slip angle, no parameter of vehicle is used, there is no error from uncertain parameter.

GPS/INS integration is post-process, and using post-processed parameter side slip angle is estimated. But to estimate state of vehicle in rear time, it is needed to develop GPS/INS integration system and algorithm in real time.

Vibration in vehicle is an important problem. In this paper, Vibration in vehicle is contained in sensor measurements. After FFT analysis about sensor measurements, vibration makes white noise in whole frequency. In this paper, smoothing is used. To develop real time system, low frequency filter is used to IMU. But that makes delay of signal and it can make problem synchronize with GPS signal. So removing sensor noise vibration from vehicle is important problem

REFERENCES

[1] Kimbrough, S, "Coordinated Braking and Steering Control for Emergency Stops and Accelerations," *Proceedings of the WAM ASME*, Atlanta GA, pp. 229-244, 1991..

[2] Kienche, U., Daiss, A., "Observation of Lateral Vehicle Dynamics," *Proceedings of the IFAC*, pp. 7-10, 1996.

[3] Alberti, V., Babbel, E., "Improved Driving Stability by Active Braking on th Individual Wheel," *Proceedings of the International Symposium on Advanced Vehicle Control*, June, pp. 717-732, 1996.

[4] Hac, A., and Simpson, M. "Estimation of Vehicle Side Slip Angle and Yaw Rate," *SAE 2000 World Congress, Detroit, Michigan, SAE* Paper No. 2000-01-0696, March 2000.

[5] R. A. Masmoudi, J. K. Hedrick, "Estimation of Vehicle Shaft Torque Using Nonlinear Observers," *ASME Journal of Dynamic Systems Measurement and Control*, Vol. 144, pp. 394-400, 1992

[6] L. R. Ray, "Stochastic Decision and Control Parameters for IVHS," *ASME IMECE Advanced Automotive Technologies*, pp. 114-118, 1995.

[7] K. Huh, J. L. Stein, "Well-Conditioned Observer Design for Observer-Based Monitoring Systems," *ASME Journal of Dynamic Systems Measurement and Control*, Vol. 117, No. 4, pp. 592-599, 1995

[8] S.P Hong, M.H Lee, J.A. Rios and J.L. Speyer, "Observability Analysis of INS with a GPS Multi-Antenna System," *KSME International Journal Vol. 16, Vol. 16, No. 11, pp. 1367-1378, 2002*

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

The Application of a Robot for the Construction of Curtain Wall in a High Building

Soungyel Lee, Bongsoo Ko
Dep. Mechatronics
Hanyang University
Seoul, 425791, Korea

Kyeyoung Lee, Sangheon Lee
Const. Equipment R&D
Samsung Corporation
Seoul, 463771, Korea

Changsoo Han
Dep. Mechanical Eng.
Hanyang University
Seoul, 425791, Korea

Abstract

Recently, the trend in architectural forms has been toward taller and larger building. The building materials, therefore, are getting larger and heavier as wall. Most of the construction projects are, however, dependent on outdated equipment and/or human resources.

Construction processes up to now face a number of problems, including dangerous work, high construction cost and heterogeneous construction quality. In various construction sites, automation in construction has been introduced to address these problems. This paper proposes a human-machine cooperative system in the construction site; the system utilizes construction of a curtain wall in tall buildings. The procedure for development of the system consists of analysis of existing curtain wall construction and design of system specifications. The aims for the development of this system include:.

- ✓ Human-machine cooperation
- ✓ Simple and precise construction procedure
- ✓ Reduction of the number of workers
- ✓ Safety assurance
- ✓ Retrenchment of the construction cost and period
- ✓ Homogeneous construction quality.

1 Introduction

Recently, the trend in building construction is toward taller and larger buildings. The building materials are, therefore, becoming larger and heavier as well. Many of the types of equipment used to handle these materials are outdated, and most of the construction work is managed by a human operator. Construction work is, therefore, fraught with a number of problems, including frequent accidents, high construction cost and heterogeneous construction quality which depends on the experience of the operator [1]. Furthermore, humankind has expended its territory. It now reaches into space and under the sea. Construction equipment, therefore, must be developed to handle these new construction challenges. To solve these problems, it is necessary to introduce automation systems [2].

A construction robot has been developed for higher productivity and better safety in many different areas of construction [3,4]. A curtain wall is one method of outer wall construction. It is appropriate for super tall buildings. For this reason, an automation system for curtain wall construction must be developed.

A curtain wall is constructed using a winch and crane, as well as many workers. Its procedure is complicated and slow; as well, it is dangerous for the workers. To improve the procedure, a mini excavator with a suction device is applied. The suction device has one rotation mechanism. Through use of the mini excavator system, a curtain wall can be moved to the assembly point easily. This system also reduces the number of workers and the amount of construction time. The curtain wall assembling, however, must still be operated by a construction worker.

The authors of this paper propose a human machine cooperative system which replaces the suction device in the mini excavator system. The proposed system has 3 DOF with a suction device. The system is controlled by human force input to use curtain wall assembly. In development of the cooperative system, the system must fit with the properties found in an actual construction environment. In particular, the following problems must be addressed:

- ✓ Since the indoor robots are working on a slab, the allowable weight of slab is limited. The weight of the system should be considered.
- ✓ The system should have sufficient workspace in order to avoid an obstacle.

This study considered these problems and conceived of a way to employ a robot in the construction of the curtain wall in tall buildings. The use of robots at construction sites can reduce the need of human involvement. Construction time and cost can be reduced as well. An

important aspect of the use of robots at construction sites is prevention of accidents [5,6].

2 Analysis of curtain wall installation

The current curtain wall construction method must be analyzed in order to design a robot. The mini excavator system is composed of a suction device and a mini excavator. A suction device holds the curtain wall with a rotation mechanism. An excavator moves the curtain wall to the assembly point.

Before analysis of the current curtain wall construction method can be conducted, a coordinate system must be defined as presented in Fig. 1. The origin of the coordinate system is located in the center of mass of the curtain wall.

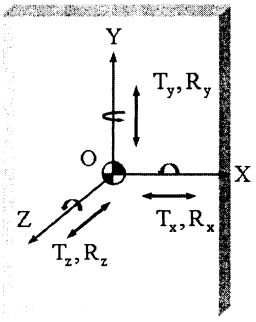
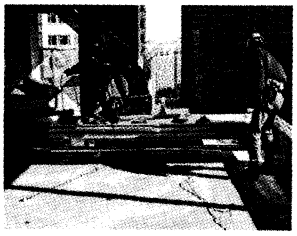
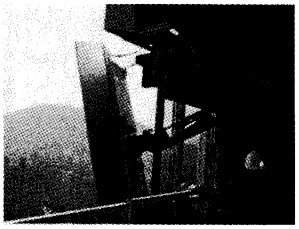


Fig. 1 A coordinate system of a curtain wall



(a) Holding

(b) Movement



(c) Adjustment

(d) Finishing (e) Completed shape

Fig. 2 The procedure of curtain wall installation

T represents translation and R represents rotation.

Fig 2 shows the current curtain wall construction procedure as described in the following:

(a) By using the suction device on the end of a mini-excavator, the curtain wall is held in place. The required DOF is R_z to align with the excavator.
(b) The curtain wall is moved toward a assembly point by a mini-excavator. The required DOF are R_z to avoid obstacles.
(c) The position and orientation of a curtain wall is adjusted by a R_z rotation mechanism. The required DOF is T_x, T_y, T_z, R_x, R_y, and R_z. The utilizable DOF in a mini-excavator is T_y, T_z, and R_x. The utilizable DOF in the rotation mechanism is R_z. A residual DOF is handled by a worker.
(d) The mini-excavator is separated from the curtain wall. Then, the final assembly work is executed by a worker. The required DOF is T_x, R_x and R_y.

3 A design of system specifications

The curtain wall construction system overview is a macro - micro motion system. An excavator is considered to be the macro motion manipulator. The system under development is considered to be a micro motion manipulator. For the macro system, any commercial equipment can be used. A mini excavator can be used in indoors; as well, it has a redundant long manipulator with sufficient power to load the curtain wall.

One of the most important considerations at a construction site is the safety of the human workers. If an excavator loads over weight, it will overturn [7]. To prevent overturning, safety tests were considered, as presented in Fig. 3. The allowable load is 500 kg. The weight of a curtain wall is 300 kg. The system under development, therefore, must weigh under 200 kg.

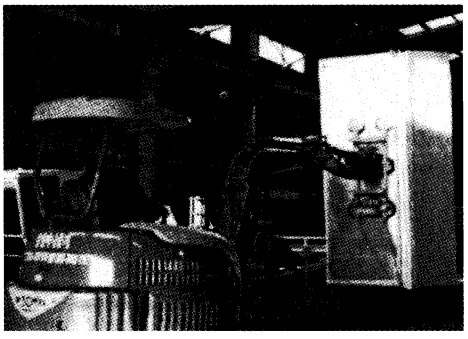


Fig. 3 Overturning test of excavator

3.1 A selection of DOF

In the construction of a curtain wall, the required DOF is T_x, T_y, T_z, R_x, R_y, and R_z. The utilizable DOF in the macro-motion manipulator is T_y, T_z, and R_x. The minimum required DOF in the micro-motion manipulator is T_x, R_y, and R_z.

The system under development can be used not only in curtain wall construction but also in other construction work. This system, therefore, is modularized to add or remove DOF.

Fig. 4 shows the diagram of the modularized design. The specifications of each module are determined by analysis of the current curtain wall construction work.

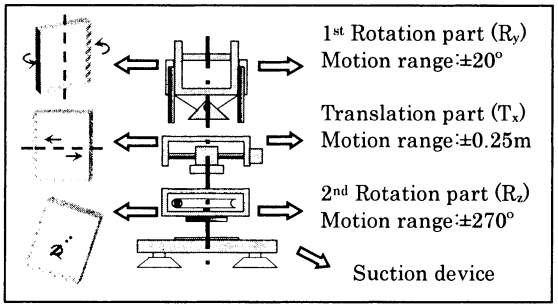


Fig. 4 The modularized design of micro-motion manipulator

3.2 The required force / torque in each module

In the design of a system, the required torque and force have to be calculated to select the proper actuator.

The required torque in the 1st module is estimated by equation (1) which is multiplies the moment of inertia and the angular acceleration. The 1st module can be simplified by two mass systems as shown in Fig. 5.

$$\tau = I_0\alpha = \left[(I_0)_1 + (I_0)_2\right]\alpha \qquad (1)$$

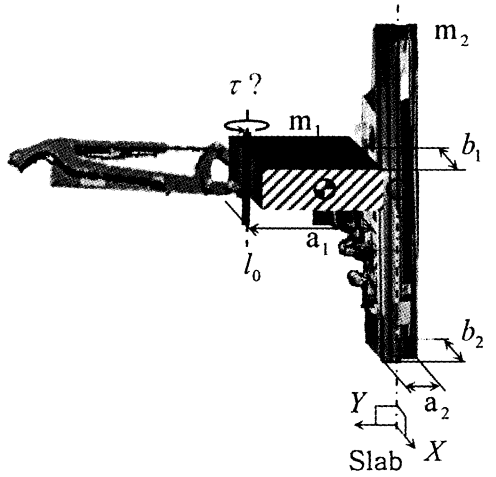


Fig. 5 The required torque of 1st module

The 2nd module consists of the translation part. If a linear guide has no friction, the required force is simply estimated by Newton's second law. It is important to note that an actuator and a reducer have to be selected with several considerations in mind. These considerations include, but are not limited to, mechanical friction and safety rate.

The required torque in the 3rd module is estimated by equation (2).

$$\tau = \tau_1\,(\text{Equilibrium}) + \tau_2\,(\text{Operation}) \qquad (2)$$

The center of a suction device is difficult to align with the center of the curtain wall mass in the real field. The maximum misalignment length is assumed to be 0.2 m. The required torque can be calculated through static equilibrium with moments of inertia of the curtain wall.

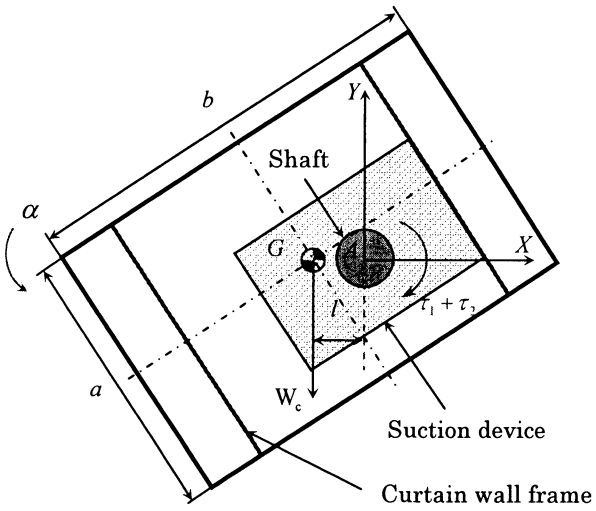


Fig. 6 The required torque of 3rd module

In table 1., there is a list of required torque/force with the operating range of each module.

Table 1. Design spec. of each module

	Required Torque/Force	Range
1st module	176.72 [Nm]	±20°
2nd module	20 [N] (no friction)	±0.25m
3rd module	725.54 [Nm]	±270°

4 Control scheme

The fully automated system is suitable for factories in the manufacturing industries. The factory can provide a controlled environment. Construction environments, on the other hand, change frequently. The fully automated system is, therefore, not suitable for construction work.

In this case, the human machine cooperative system is more suitable for the construction task. Fig. 7 shows the control scheme of the curtain wall construction system [8]. An operator provides commands to the micro manipulator with force input. An operator can feel the reaction force from the curtain wall [9].

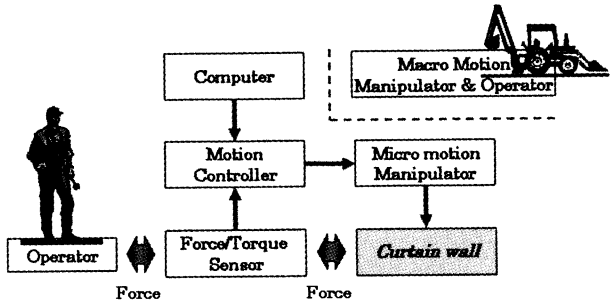


Fig. 7 Control scheme of the curtain wall construction system

The concept of the system under development is presented in Fig. 8.

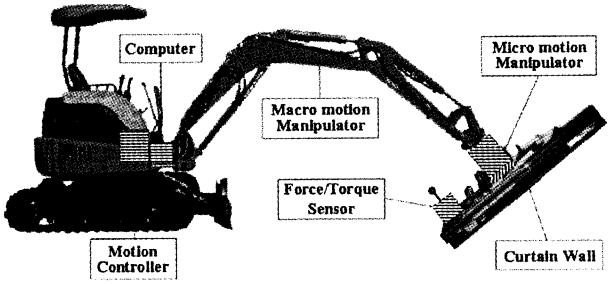


Fig. 8 Concept of the developing system

5 Conclusions

The ultimate goal of the proposed system is for human-machine cooperation. The robot is commanded/operated through human force; in turn, it assists the human operators [10].

The system described in the present study is one step on the way to full automation. There is still a long way to go; however, the research and development continues. The advantages of proposed system include the following:

- ✓ Simple and precise construction procedure
- ✓ Reduction of the numbers of human workers
- ✓ Safety assurance
- ✓ Retrenchment of the construction cost and period
- ✓ Homogeneous construction quality

Much of the automation in construction has been developed for outdoor work. The proposed robot is being developed for curtain wall construction work at indoor sites. The micro-motion manipulator is modularized to allow for addition or removal of a DOF. Furthermore, the proposed robot can be applied to other construction work by exchanging the end-effecter.

References

[1] H.J. Sim ; C.S. Han, "The Development of a Robot Hand for the Automation of Steel Column Construction," *IFAC Symposium on Robot Control - Syroco,* pp. 723~728, 2000

[2] Albus, James S. "Trip Report: Japanese Progress in Robotics for Construction," *International Journal of Robotics*, Vol. 2, No. 2, pp. 103~112, 1986

[3] Roozbeh Kangari, "Advanced Robotics in Civil Engineering and Construction," *91 ICAR., Fifth International Conf*, Vol.1, pp. 375 ~378, 1991

[4] A. Warszawski, "Economic implications of robotics in building," *Building and Environment*, Vol.20, Issue 2, pp. 73~81, 1985

[5] Cusack, M, "Automation and Robotics the Interdependence of Design and Construction Systems," *Industrial Robot,* Vol.21, No.4, pp.10~14, 1994

[6] Wen, Xia ; Romano, V.F. ; Rovetta, A., "Remote Control and Robotics in Construction Engineering," *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on 1991,* pp. 1429~1432, 1991

[7] O.J. Kim ; W.S. Yoo ; K.H. Yoon ; H.G. Kang, "Evaluation of Joint Reaction Forces for a Hydraulic Excavator Subjected to a Critical Load," *KSME,* Vol.20, No.4, pp. 1154~1163, 1996

[8] Manfred Hiller, "Robotics and Autonomous Systems : Modeling, simulation and control design for large and heavy manipulators," *Robotics and Autonomous Systems*, Vol.19, Issue 2, pp. 167~177, 1996

[9] Inaba, T. ; Hayashizaki, M. ; Matsuo, Y., "Design of a human-machine cooperation system to facilitate skilled work," *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on 1999*, Vol.4, pp. 995~1000

[10] Manfred Hiller, "Robotics and Autonomous Systems : Modeling, simulation and control design for large and heavy manipulators," *Robotics and Autonomous Systems*, Vol.19, Issue 2, pp. 167~177, 1996

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Fingerprint Classification Method Using Measurement of Ridge Energy Changes

Yoon KyungBae
Dept. Computer Science & Engineering
Kimpo College, Kimpo, Korea
kbyoon@kimpo.ac.kr

Park ChangHee
Dept. Electronics Engineering
Yonsei University, Seoul, Korea
chpark92@yonsei.ac.kr

Abstract

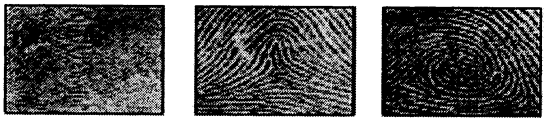
The existing classification system, Henry System, is useful in a captured fingerprint image of core point and delta point using ink-roll. However, the Henry System is unapplicable in modern Automatic Fingerprint Identification System (AFIS) because of the problems such as size of input sensor and way of input. In a modern Automatic Fingerprint Identification System (AFIS), Henry System, a traditional fingerprint classification model, is difficult to apply. To tackle this problem, this study is to apply algorithm for an An Ensemble Fingerprint Classification Method using Measurement of ridge energy changes in order to improve precise matching speed of a large volume of database.

Keyword: Biometrics, Fingerprint Classification, Fingerprint Identification, Singular Point, Minutia

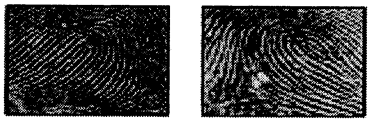
1. Introduction

Fingerprint is an wrinkle-shaped image formed from ridge of perspiratory gland and has characteristics that everyone has different shape and no change in his or her life. It has been used very often as an individual recognition device since Edward Henry did a systematic research in the mid 19 century. Moreover, the safe share and security of personal information is increasingly of importance. The research of fingerprint recognition skills is being conducted vigorously to solve the problems of loss, forgetfulness, and illegal use[1].

The Characteristics of fingerprint are generally divided into S ingular Point and Minutia. The Singular consists of a core point and a delta point while the minutia is made up of a straight point and a diverging point. Henry Classification System is to classify 5 patterns of configurations by using information of singular point of a core point and a delta point. a meager face is without core and delta points, an upward meager face fingerprint is with a core point but without a delta point, a right-side sacrificed table fingerprint with a core point and w ith a delta point in left-side, a left-side sacrificed table fingerprint is with a core point and a delta point in right-side, and a whorl fingerprint is with two core points and two delta points[2][3][10].



(a) Arch (b) Tented Arch (c) Whorl



(d) Left Loop (e) Right Loop

Fig. 1 Basic 5 patterns of fingerprint

Figure 2 shows a normal process of fingerprint classification. First, Fingerprint image is input and then whole processes of noise deletion and preprocessing of segmentation orientation are completed. Third, after detection of a core point and a delta point classification is done using numbers and positions of a core point and a delta point[3].

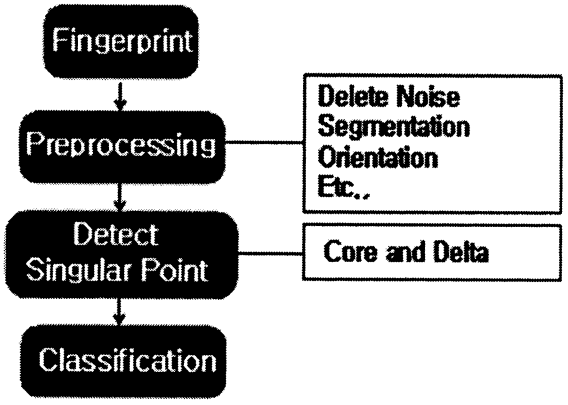


Fig. 2 Normal Process of Fingerprint classification

Figure 3 shows a fingerprint image and minutia captured by a printing method signing with turn on fingerprint paper and ink and Figure 5 shows captured fingerprint image and minutia using semiconductor sensor.

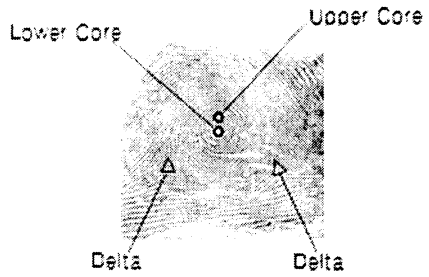


Fig. 3 Captured fingerprint image and minutia using paper and ink

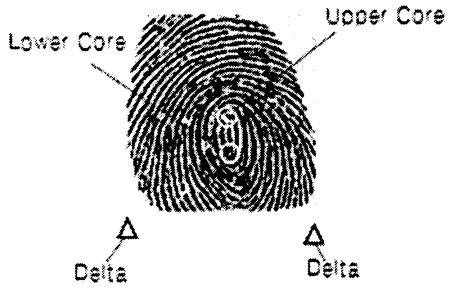


Fig. 4 Captured fingerprint image and minutia using semiconductor sensor

This study is aimed at Fingerprint Classification System which can classify 5 basic patterns of Henry System in uncaptured delta image using changes of gradient of ridge.

2. Method

In this study, fingerprints are offered through a semiconductor sensor and an optical sensor. The input fingerprints carry out elimination of noise and separate a fingerprint image from a background image through segmentation. And through orientation a direction component is detected and find out a core point by using Poincare Equation[2][3]. Even a meager face without a core point spots the place which has the largest degree of leaning and regards it as a core point.

2.1. Least Entropy

Figure 5 is an explanation map which sets up main orientation of a raised-line searching for the least entropy from a center point. If the entropy value of a certain region from a core point is measured and its least valued orientation is found, the orientation matches with orientation of a raised-line.

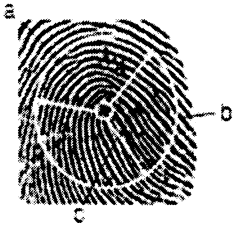


Fig. 5 Least entropy from center point

In Figure 6, if the brightness of pixel is set as P(n), and the length of a radius is L, Entropy value is calculated from a core point to a straight line(c) along circumference of a circle(b), which produces a formula(1).

$$\frac{1}{L}\sum_{n=0}^{L} P(n) \qquad (1)$$

Entropy value in specific area is earned from this acquired core point and the main orientation is made for the value to be least. The established main orientation has the least crossing of Ridge protraction line and ridge, and is established toward the way of protraction line.

2.2. Orientation of Region

Figure 6 shows the establishment of the main orientation, the least entropy value from a core point and then recalculates local region orientation from a certain distance away vertically and right-left symmetrically along the orientation and set up the inclination of volume of variation. At this moment, The size of blocks must be twice an average distance between raised lines, find orientation components and determine the orientation component from the center of the blocks.

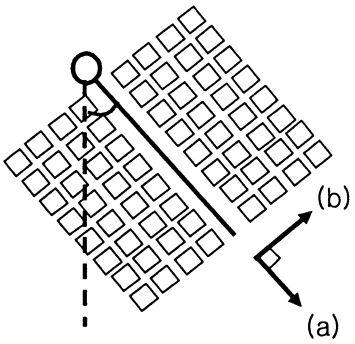
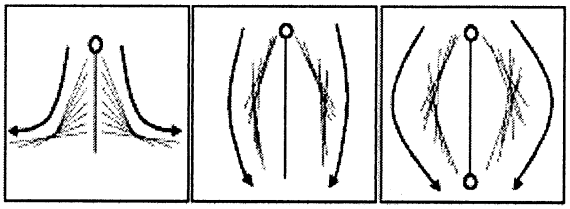


Fig. 6 Local region orientation

If each inclination of blocks is calculated, the amount of right-left side change in inclination of a main orientation is measured, and then common inclination areas and changes of 5 basic patterns are measured and at first are classified into Arch, Loop, Whorl. In case of Right Loop which is to calculate a main orientation, determine Left Loop or Right Loop using the acquired orientation inclination. An Arch is to get distance to vertical changes of a core point and inclination and is classified as an Arch, a Tented Arch when its value is below and above, respectively.

Figure 7 is a generalized picture with fingerprint shapes of change pattern according to blocks of local region orientation established in Figure 6.



(a) Arch (b) Loop (c) Whorl

Fig. 7 Patterns of change of gradient

2.3. Measurement Ridge Energy Changes

Figure 8 represents generalized patterns of change of gradient after regional inclination calculation. At this point, the standard of classification measures variation volume of inclination between α and β and is classified as Arch(a), Loop(b) and Whorl(c).

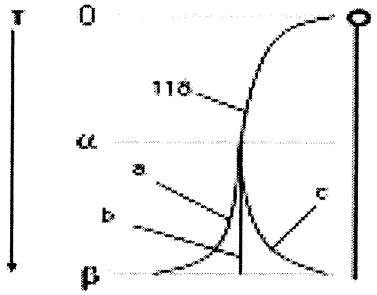


Fig. 8 Patterns of ridge Energy Changes

The amount of common inclination change of gradient is shown as formula (2). This represents the changes of inclination from 0 to α.

$$\delta \cong \frac{\pi}{2\alpha} T \quad (0 \leqq T \leqq \alpha) \tag{2}$$

After that, as for the amount of inclination change, Arch produces formula (3), Loop shows (4) and Whorl makes formula (5).

$$\delta \cong -\frac{\pi}{2(\beta-\alpha)} T \quad (\alpha < T \leqq \beta) \tag{3}$$

$$\delta \cong \frac{\pi}{2} \quad (\alpha < T \leqq \beta) \tag{4}$$

$$\delta \cong \frac{\pi}{2(\beta-\alpha)} T \quad (\alpha < T \leqq \beta) \tag{5}$$

In the above formula, T represents the distance moving along a central axis, also δ is showing inclination 1 functional formula. α is the distance representing similar inclination according to is pattern, β is the distance representing different inclination according to its pattern. the common region is formula (6) by differential equation. In case of Arch, formula (7), in case of Loop, formula (8) and in case of Whorl, formula (9).

$$\frac{d\delta}{dT} \cong \frac{\pi}{2\alpha} \quad (0 \leqq T \leqq \alpha) \tag{6}$$

$$\frac{d\delta}{dT} \cong -\frac{\pi}{2(\beta-\alpha)} \quad (\alpha < T \leqq \beta) \tag{7}$$

$$\frac{d\delta}{dT} \cong 0 \quad (\alpha < T \leqq \beta) \tag{8}$$

$$\frac{d\delta}{dT} \cong \frac{\pi}{2(\beta-\alpha)} \quad (\alpha < T \leqq \beta) \tag{9}$$

3. Classification Rule

When the number of core points(Cn) is 2 after confirming the number of core points, the change of gradient(△δ) is measured along orientation established from a central axis. If it fits the condition of Whorl, it is classified as Whorl. If not, another fingerprint is accepted.

When the number of core points(Cn) is 1 the change of gradient(θc) is measured along orientation established from a central axis. If it fits the condition of Loop, it is classified as Loop. And if it is above 0, it is classified Right Loop, If it is below 0, it is classified Left Loop. If the change of gradient(△δ) satisfies the condition of Arch, and the distance from a core point(Lc) is shorter than the distance(Tc) gotten from a critical point, it is classified Arch. If it is far, it is classified tented Arch. If the core point is 1 but change of gradient fits the condition of Whorl, it is classified as Whorl. This means that because of big width of core points fingerprint sensor can not receive the below core point.

Figure 9 shows the block diagram of fingerprint classification method using measurement of ridge energy changes designed in this study.

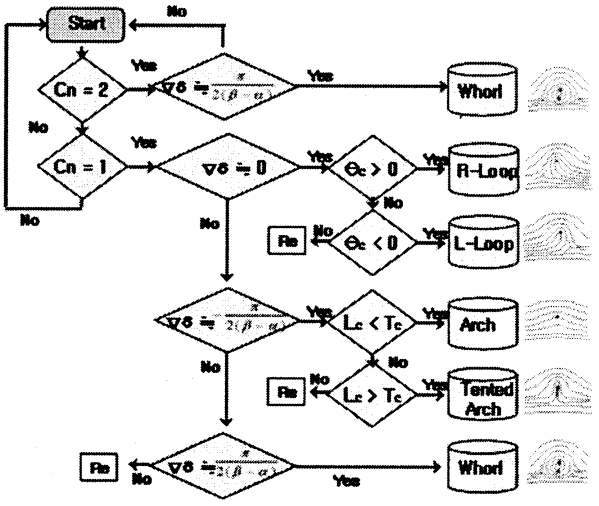


Fig. 9 Determine the class of fingerprint

4. Experimental Result

The experiment of this study is to construct a volume of fingerprint database and as a first step by it is confirmed that 5 patterns of Arch, Tented Arch, Left Loop, Right Loop, Whorl through Henry System. This study is to suggest a Fingerprint Classification System which can classify 5 basic patterns of Henry System in uncaptured delta image using changes of gradient of ridge. This fingerprint classification technique will make an improvement of precise matching speed by reducing data volume. Through this experiment it is revealed that recognition speed is improved in case of more fingerprints data than average. The suggested fingerprint classification algorithm used fingerprint of 364 students of A College, using semiconductor sensor of Authentec Company. It constructed a database of 3,640 from 10 fingers per person. Fingerprint image uses 256×256 size, 8 Bit Gray image. 3% of fingerprint image, 109, can not be classified because they are low-quality and invisible owing to sweat and other substance. 97% of fingerprint image, 3,531, conform to 5 patterns of classification and the conformity result reached 91.3.

Table 1. Result of fingerprint

Test / True	Arch	Tented Arch	Left Loop	Right Loop	Whorl
Arch	**66**	5	3	2	2
Tented Arch	3	**29**	2	3	2
L-Loop	33	26	**941**	16	20
R-Loop	16	22	10	**688**	6
Whorl	25	4	32	15	**1,554**

The fingerprint matching experiment applied MST (Minimum Spanning Tree) algorithm, and its application method is to increase gradually the number of database as 10, 100, 500, 1,000, 2,000, 3,000 and to measure the time taken.

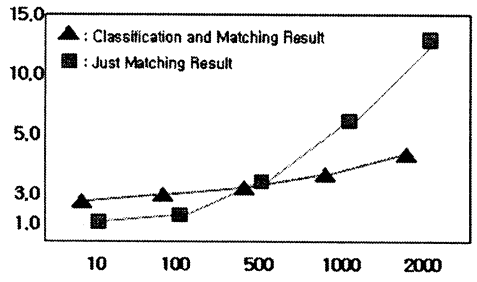


Fig. 10 Comparison matching result

5. Conclusion and Future Work

This study is to suggest an Ensemble Fingerprint Classroom System which can classify 5 basic patterns of Henry System in uncaptured delta image using changes of gradient of ridge. This fingerprint classification technique will make an improvement of precise matching speed by reducing data volume. Through this experiment it is revealed that recognition speed is improved in case of more fingerprints data than average.

In the future research, in order to apply classification algorithm with more speed and accuracy, the better results with excellent performance will be provided from statistic learning theory in combination with neuro network theory.

6. Acknowledgement

This research was supported in part by a grant from the 2003 Kimpo College.

7. References

[1] M. Chong, T. Ngee, L. Jun, R. Gay, " Geometric framework for fingerprint image classification, " Pattern Anal. and Machime Intell., Vol 19(4), pp 302-304, 1997.

[2] Anil Jain, Lin Hong, Ruud Bolle, " On-Line Fingerprint Verificaiton, " IEEE Transactions on pattern analysis and machine intelligence, Vol. 19, No. 4, Apr 1997.

[3] Lin Hong, Anil Jain, " Introduction to biometrics, " Biometrics-Personal Identification in Networked Society, KluweAcademic Publishers, pp. 1-41, 1999.

[4] C. L. Wilson, G. T. Candela, C. I. Watson, " Neural-network fingerprint classification, " pp. 203-228, 1994.

[5] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. " Fingerprint classification by directional image partitioning, " Transactions on Pattern Analysis Machine Intelligence, Vol 21(5), pp 402-421, 1999.

[6] J. Platt, N. Cristianini, and J. Shawe-Taylor, " Large margin dags for multiclass classificatin, " In Adcances in Neural Information Processing Systems, Denver, Colorado, 2000.

[7] S. Ando, " Consistent Gradient operators, " IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 20, pp. 252-265, 2000.

[8] Maria Petrou and Panagiota Bosdogianni, " Image Processing : The Fundamentals, " John Willey & Sons Ltd, 2000.

[9] L. C. Jain, U. Halici, I. Hayashi, S. B. Lee and S. Tsutsui, " Intelligent Biometric Techniques in Fingerprint and Face Recognition, " CRC Press, 1999.

[10] Q. Zhang and H. Yan, " Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudo Ridges, " submitted to Pattern Recognition in Feb. 2002.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Generating a Virtual Human Model for a Mixed Reality Space

I. Yamaguchi, J. K. Tan, S. Ishikawa

Department of Mechanical and Control Engineering, Kyushu Institute of Technology,
Sensuicho 1-1,Tobata, Kitakyushu 804-8550, JAPAN

Abstract

The concept of virtual reality has gained its popularity among people. It enables us to be in a virtual space where we can get information on what is not easy to obtain in our real space. Virtual reality has then produced the notion of mixed reality. The technique of mixed reality lets us place virtual entities in a real world. We have been developing techniques for creating a 3-D virtual human and putting it into a real space.

This paper describes a technique for generating 3-D models of arbitrary human motions from a set of primitive motions obtained by a motion capture technique. Some human basic motions are modeled in a 3-D way by motion capture as primitive motions and they are combined into a new synthesized motion sequence by the application of rotation, parallel translation and linear interpolation to the primitive motions. Experimental results show satisfactory performance of the proposed technique.

1 Introduction

Virtual reality is a technique for making a virtual space in which we can walk around to observe virtual objects that compose the virtual space. The concept of virtual reality has already obtained its popularity among people in various application fields. Virtual reality has then produced the notion of mixed reality [1], which is a technique for making virtual entities appear in a real world. We have been developing a technique for creating a virtual human model that moves and acts as we do, and making it come on a real space in front of a person through a see-through type head mounted display [2,3]. In our goal, the virtual human will communicate with a user and give him/her various kinds of information convenient for living life. It may therefore have various application fields including a virtual human-assisted instruction system in machines operation, a virtual human-assisted route guiding system, *etc.*

This paper describes a technique for generating 3-D models of arbitrary human motions, given a set of primitive motion models obtained by motion capture. Some basic human motions including walking, standing up, sitting down, *etc.*, are modeled in a 3-D way by motion capture as primitive motions. They are then connected smoothly into a new synthesized motion sequence by applying rotation, parallel translation and linear interpolation to the primitive motions. The motion capture system employed in this study is the one originally developed by Tan & Ishikawa [4,5]. Unlike other established techniques based on stereo vision [7,8,9], it employs uncalibrated cameras for video image taking. It can therefore be used in outdoor environment without difficulty as well as indoor circumstances.

In the performed experiments, some primitive motions are combined into a single synthesized motion. The primitive motions include walking, standing up, sitting down, and raising a hand. Experimental results are shown and the results are discussed.

2 A Motion Capture System

The employed motion capture technique is our original technique using an extended measurement matrix and its decomposition [4,5]. A factorization method proposed by Tomasi & Kanade [6] employs a measurement matrix and its singular value decomposition to perform 3-D shape recovery. The measurement matrix contains the xy image coordinates of all the feature points specified on a rigid object interested. On the other hand, our extended measurement matrix contains not only spatial but also temporal positions of the feature points on images. Therefore the extended measurement matrix can describe deformation of nonrigid objects.

In the employed motion capture system, more than three cameras are fixed around an object concerned. Let us denote a matrix $W(t)$ (t=1,2,...,T) whose entries are the xy image coordinates of the feature points specified on an object. Parameter t is observation time. We define an extended measurement matrix W in the following form;

$$W = \left(W(1)\, W(2) \ldots W(T)\right). \qquad (1)$$

The number of columns of submatrix $W(t)$ represents the number of feature points at sample time t and they

are usually independent with each other for t=1,2,...,T. This means that feature points can appear or disappear at each sample time t.

Matrix W is decomposed into two matrices by singular value decomposition as follows;

$$W = MS, \tag{2}$$

where matrix M, called an orientation matrix, gives camera orientation, whereas matrix S, called a shape matrix, contains 3-D coordinates of all the feature points registered in W. Thus the 3-D deformation process of a nonrigid object recovers by the matrix S.

Advantages of the employment of the extended measurement matrix given by Eq.(1) include that all the deformation process of a nonrigid object recovers simultaneously without 3-D data alignment along the time axis, and that random noises contained in the x and the y coordinates described in W are well averaged because of a large number of columns in W, resulting in less recovery errors. It may be added that calibration is not necessary with the employed cameras in our system. This is because factorization [6] is applied to matrix W in order to recover 3-D shape. The recovery process is actually far much different from the direct linear transformation (DLT) method [7,8] normally adopted for 3-D recovery in most products on 3-D recovery or motion capture.

Since the coordinate system in which the feature points are defined has its origin at the average point of all the considered feature points in a real space, it is not an absolute coordinate system. This means that the coordinate system is independently defined each time 3-D shape recovery is done employing Eqs.(1) and (2).

Configuration of the entire system is illustrated in **Fig.1**. $F(\geq 3)$ cameras are placed fixed around the object concerned. Their image streams are fed into a PC either on-line or off-line. Once feature points are tracked to yield the matrix W given by Eq.(1), calculation for 3-D shape recovery is done and the resultant 3-D model is displayed. It is noted that, if more than six cameras are placed around the object concerned, its entire shape and motion recover at a time [10], although the cameras are placed in front of the object in the performed experiment.

3 Synthesizing Motions

The motion capture technique stated in the former section is employed for making 3-D models of human motions. It is natural to connect primitive motions in order to represent arbitrary human motions. To realize it, each primitive motion should be somehow adjusted so that it can be connected smoothly to a successive primitive motion. This is not very simple, however, when motion capture data has the description form of a set of 3-D coordinates of the feature points specified on a human body, since the detected number of feature points,

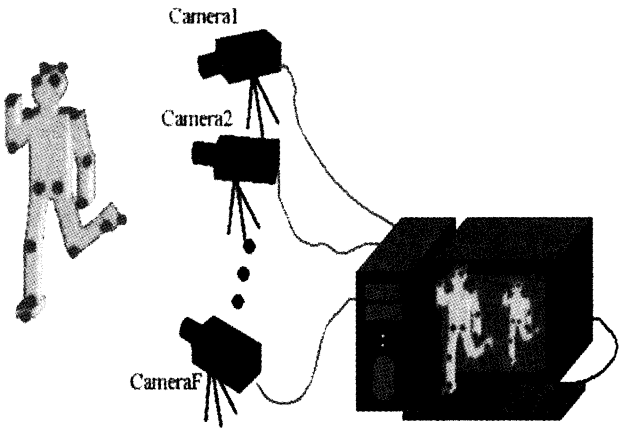


Fig.1. Configuration of the system.

and the location and the scale of the coordinate system often depend on experiments.

The proposed method realizes seamless connection first by superposing identical positions between two motions to be connected by the employment of rotation and translation, and then by performing interpolation between the successive frames composing the connected part of the motions.

In the employed motion capture technique, one cannot specify the 3-D coordinate system in which locations of recovered feature points are described. This means that two motions obtained from individual 3-D shape recovery are described in the respective 3-D coordinate systems with possible scale difference. Therefore transformation between the two coordinate systems is indispensable to realize seamless connection of successive motions. By the seamless alignment of individual motions, a model of arbitrary motion can be produced.

Employed format of the 3-D recovered data is the 3-D coordinates. By these data, a transformation matrix is defined containing translation, rotation, and scaling in order to superpose one coordinate system to the other. If the number of recovered feature points is mutually different between two recovered motions, the coordinates of the missing feature points are interpolated in a 3-D way to make the number of the feature points identical. To realize further smoothness, two successive motions are interpolated between the end of the first motion and the beginning of the successive motion. Linear interpolation is employed for this purpose.

4 Experimental Results

In the performed experiment, the entire system is composed of three fixed cameras and a PC. Motions of a person are taken images by the three cameras and the image streams are fed into a PC for further processing.

The central view of a subject taken by the central camera is shown in **Fig.2**. In the figure, a person wears black clothes on which color markers are adhered. As shown in the figure, motion images have been taken outdoors. This is for demonstrating that our motion capture technique doesn't necessitate camera calibration and it is particularly advantageous for outdoor video image taking.

The number of feature points adhered on the person is 18. The obtained video images are stored into a memory as a set of still images containing 24 frames per second. From these stored images, the extended measurement matrix W of Eq.(1) is defined and, by applying the singular value decomposition to W, the shape matrix S is obtained. Computation time for recovering a 10 second motion (240 frames) with 18 feature points is approximately 5 seconds by a PC equipped with a Pentium III processor (1.2GHz) as a CPU. These numerical data apply to both the standing up motion and the raising the right hand motion.

In the performed experiment, three motions, *i.e.*, standing up from the chair, raising a hand, and sitting down on the chair, are taken into consideration as primitive motions. Initially the respective motions recover three-dimensionally in separate experiments using the developed motion capture technique. They are then applied the seamless connection. The synthesized motion is given in **Fig.3**. The result was satisfactory. In the synthesized motion, a human model stood up from a chair and raised his right hand and then he sat down on the chair without loosing smoothness of the entire motion. Since this is a 3-D model, one is able to observe the motion from an arbitrary orientation.

5. Discussion and Conclusions

This paper proposed a method of synthesizing a human motion from some primitive motions given as 3-D models. The intention of the synthesis was to make arbitrary human motion models and to superpose them into a real space employing the technique of mixed reality. Some experimental results were shown in the paper, which realized seamless synthesis of a few primitive motions.

The proposed process for the synthesis has not yet been fully automated especially in defining parameters for rotation, parallel translation and scaling. This is going to be realized in an automated way in the next stage of the study.

A virtual human model shown in Fig.3 is a very simple model and it needs to be made as much as similar to a human. We have used a powerful CG software product for producing a 3-D human model [2], but it still exhibits unnatural movement. We are planning to employ an avatar to represent motions better.

The proposed technique may contribute to the production of 3-D models of various motions from prepared primitive motions. This may mean that the method imposes a user less laborious work of motion capture, since he may make use of other 3-D motion models already recovered by separate experiments.

The developed technique will be employed for creating a 3-D virtual human that can move and act as a human does. It will then be made appear in the real space in front of a user through a see-through head mounted display. One of the most useful applications of this system may be a man-machine interface. A virtual human interactively communicating with a user will be able to give various information that will make our daily life much more convenient and satisfactory. Examples may include operation explanation of various instruments by 3-D CG character, and a guide system by a virtual character taking a guest to his/her destination using GPS, *etc.*

This study was in part supported by JSPS. KAKENHI(14580450), which is greatly acknowledged.

References

[1] U. Ohta, H. Tamura(Eds.): *Mixed Reality*, Ohmsha, Ltd.(1999).

[2] D. Hirohashi, J.K. Tan, S. Ishikawa: "Producing a virtual object with realistic motion for a mixed reality space", *Proc. of Int. Conf. on Control, Automation and Systems*, pp.1084-1087 (Oct., 2001).

[3] I. Yamaguchi, K. Tou, J. K. Tan, S. Ishikawa, T. Naito, M. Yokota: "Generating 3-D models of human motions by motion capture", *Proc. of Int. Conf. on Control, Automation and Systems*, pp.1629-1632 (Oct., 2003).

[4] J. K. Tan, S. Ishikawa : "Human motion recovery by the factorization based on a spatio-temporal measurement matrix", *Computer Vision and Image Understanding*, Vol.82, No2, pp.101-109 (2001).

[5] Tan, J. K., Ishikawa, S.: "Recovering 3-D motion of team sports employing uncalibrated video cameras", *IEICE Trans. on Information and Systems*, Vol.E84-D, No.12, 1728-1732 (2001).

[6] C. Tomasi, T. Kanade: "Shape and motion from image streams under orthography: A factorization method, *Int. J. of Computer Vision*, Vol.9, No.2, pp.137-154 (1992).

[7] Y.I. Abdel-Aziz, H.M. Karara: "Direct linear transformation from comparator coordinates into object space coordinates in close range photogrammetry", *Proc. Amer. Soc. Photogrammetry*, 1-19 (1971).

[8] R. Shapiro: "Direct linear transformation for three-dimensional cinematography", *Res. Quart.*, Vol. 49, 197-205 (1978).

[9] R. C. Gonzalez, P. Wintz: *Digital Image Processing*, Addison-Wesley (1987).

[10] J.K. Tan, S. Ishikawa: "A method of modeling deformation of an object employing surrounding video cameras", *International Archives of Photogrammetry and Remote Sensing*, XXXIII, Part B5, 802-808 (July, 2000).

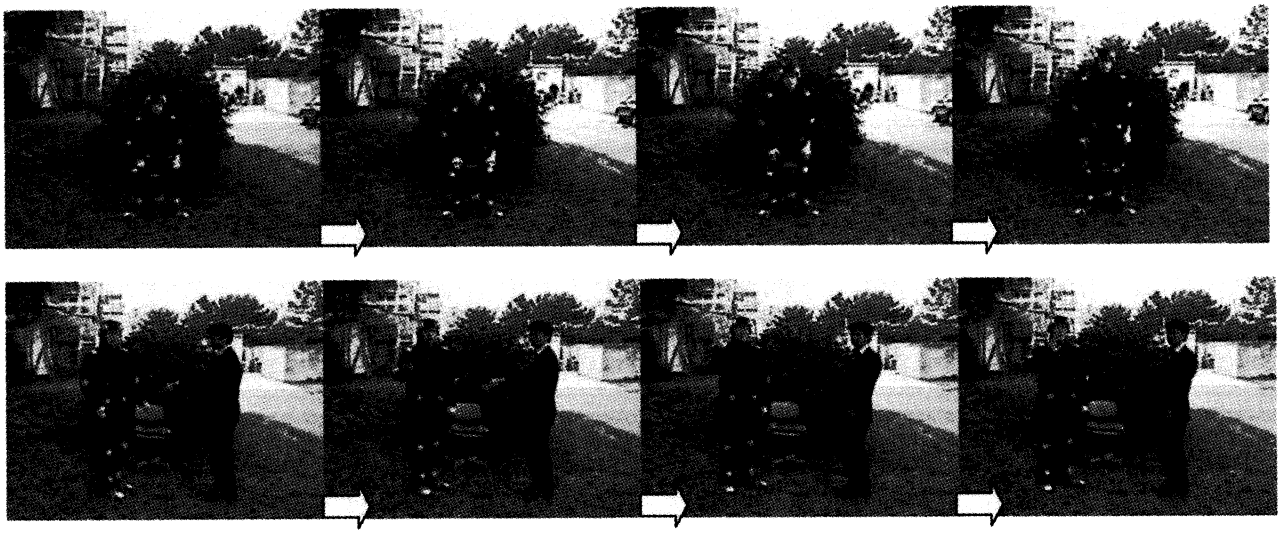


Fig.2 Primitive motions. The upper low shows a standing up motion. The lower low illustrates a raising a hand motion (See the left person). Time proceeds as indicated by arrows.

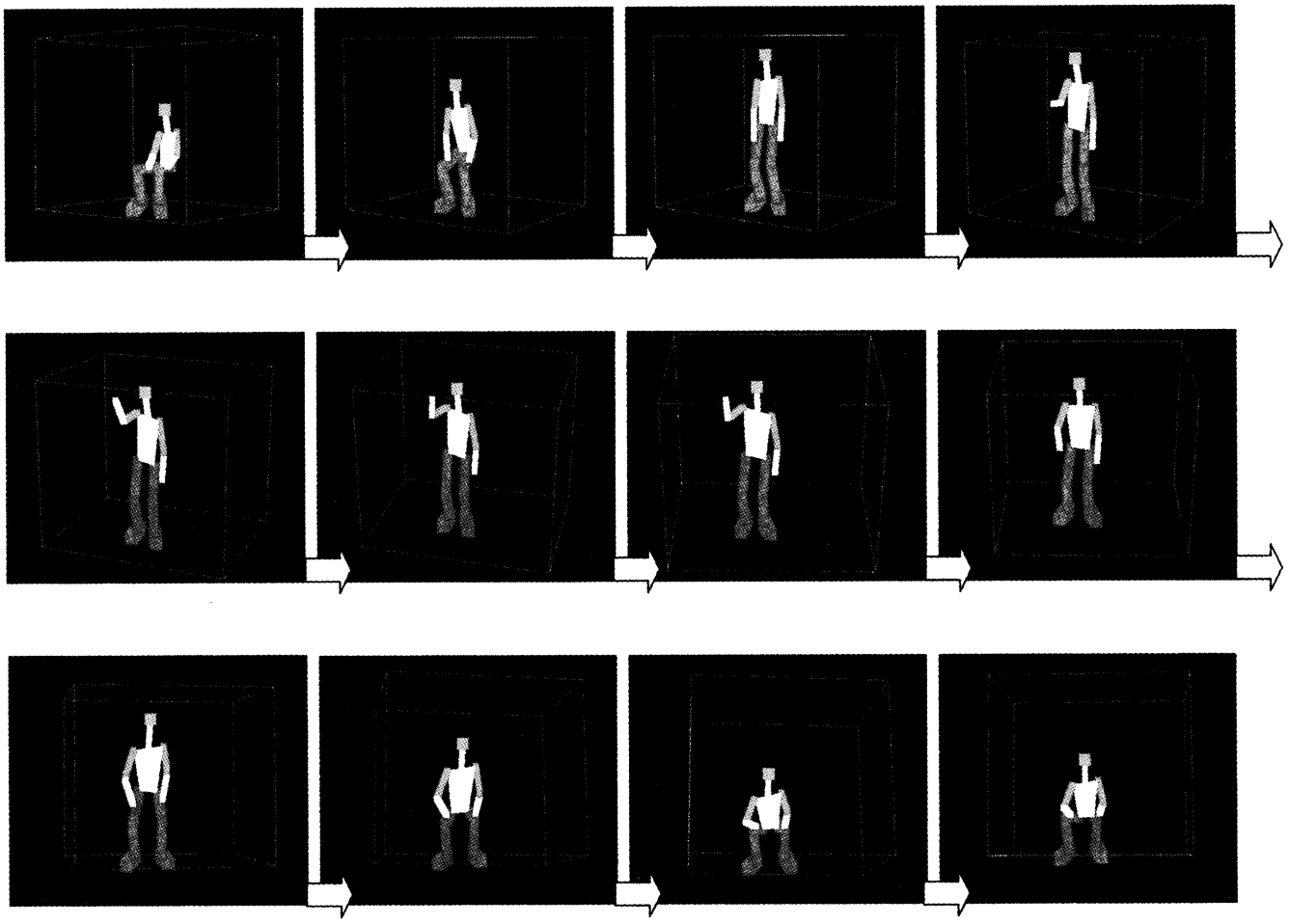


Fig.3 Synthesized 3-D motion. A person stands up (the top low), raises his right hand (the middle row), and sits down (the bottom low). Time proceeds from left to right and from top to bottom.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

In-Vitro Experimental Results of Real-time Visual Servoing for the Laparoscope Manipulator

Jin-Seok Heo
Dept. of Mechanical Engineering
Korea Advanced Institute of
Science and Technology,
Taejon, Korea 305-701
dandyheo3070@kaist.ac.kr

Min-Seok Kim
Dept. of Mechanical Engineering
Korea Advanced Institute of
Science and Technology,
Taejon, Korea 305-701
kithnkin@kaist.ac.kr

Jung-Ju Lee
Dept. of Mechanical Engineering
Korea Advanced Institute of
Science and Technology,
Taejon, Korea 305-701
jjlee@mail.kaist.ac.kr

Abstract

In this paper, we present the experimental results using the newly developed real-time visual servoing unit for laparoscopic surgery. This servoing unit is composed of the two-stage adaptive CONDENSATION (conditional density propagation) algorithm to detect the accurate position of the surgical tool tip from a surgical image sequence in real-time and the virtual spring system to control a laparoscope manipulator. According to the in-vitro experimental results, some situations such as abrupt changes of illumination and surgical tools which are frequently happened in the procedures of laparoscopic surgery have no effect on this real time visual servoing algorithm, and the proposed visual servoing unit operates the AESOP™ manipulator in real-time (25frames/sec) aiming to the tracking target. Therefore, this visual tracking algorithm is highly robust to the surgical situations and the controlled AESOP™ manipulator will be able to supply the stable view to an operator.

1. Introduction

In conventional minimally invasive surgery, surgeons operate with long, thin instruments through 'keyhole' incisions approximately 10 mm in diameter in the abdomen of the patient. An endoscope, a thin optical tube which is inserted through one of the incisions and connected to an external video camera, is used to visualize the internal organ structures and instrument tips. The endoscope video camera image is displayed on a monitor during surgery. Since a single surgeon generally has both hands occupied with surgical instruments (for example, one for grasping and one for cutting), an assistant is necessary simply to hold the endoscope steady in a desired position.

Several robot systems have previously been developed for laparoscopic endoscope manipulation during surgery [1, 2, 3, 4]. Commercially available robotic surgical endoscope manipulators are included in the da Vinci [5] system from Intuitive Surgical Systems and Aesop [6] and Zeus from Computer Motion.

The primary motivation for our endoscope manipulator project is to develop an unobtrusive, robust robotic assistant to position the surgical endoscope during surgery instead of occupying an additional trained surgeon.

To meet these needs we have aimed towards an automatic controlled manipulator. Several methods of obtaining the position information of tool tip in surgery have been suggested. Among those methods, the visual tracking is an efficient one, because it needs not additional sensor system but the information of surgical images.

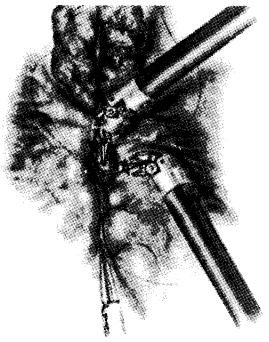


Figure 1 Laparoscopy surgery (MIS)

Casals, A [7] and Cheolwhan Lee [8] used shape information of surgical tool as a feature of tracking target. This method detects edge-information and then verifies the result with pre-defined tool shape. This method is a reliable one but it may not work well in the case that the some part of tool is covered by obstacles. It is a time consuming and computing resources consuming method, and it is hard to satisfy the real-time tracking. Guo-Qing Wei [9] and Omote, K [10] use color feature of artificial mark which is attached to surgical tool. This method has simpler structures than those of the previous method, so it costs less computational resources. However it requires additional task, attaching artificial mark. It has many problems such as choice of attaching method, and sterilization of mark. In addition, surgeons' concern is not on the mark but on the surgical tool tip, so tracking failure can be happened when the mark is covered by obstacles or it is out of image. In this paper, we suggest a light weight and simple structure manipulator with a robust visual servoing unit using CONDENSATION visual tracking algorithm.

We choosed the real time servoing system to control the manipulator using two staged CONDENSTION and virtual damping algorithm [11]. To control laparoscope automatically, controller must have information about the position of tool tip in surgical operation. We suggested new visual tracking algorithms. Our method is based on the CONDENSATION algorithm which detects the color information of image. We modified the CONDENSATION to adapt the operation situations which are very frequently happened such as the changes

of intensity of illumination and surgical tools. And then we apply the modified CONDENSATION tracker to two stages. Once the tool tip position is obtained by the tracking part, the control part of real time visual servoing unit sends the command signal for the laparoscope to be aimed to the tracking target which is located in the center of surgical image using virtual spring algorithm.

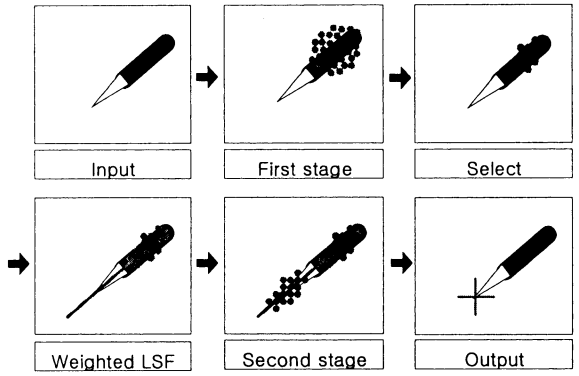


Figure 2 Two stage adaptive CONDENSATION

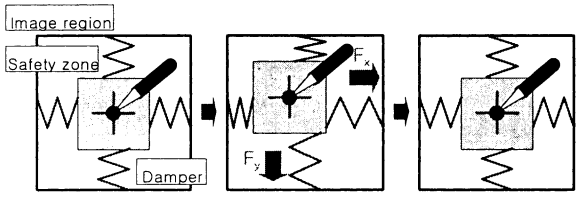


Figure 3 Virtual spring control strategy

In this paper, we present the experimental results using the newly developed real-time visual servoing unit for laparoscopic surgery which is our previous work. In-vitro experiment is performed in the stomach of dummy which has similar environmental colors. Experiments have two purposes which are to check the adaptability of this system in some situations such as abrupt changes of illumination and surgical tools which are frequently happened in the procedures of laparoscopic surgery and to verify the tracking speed of this visual tracking unit to confirm the real-time procedure.

2. Experimental Setup

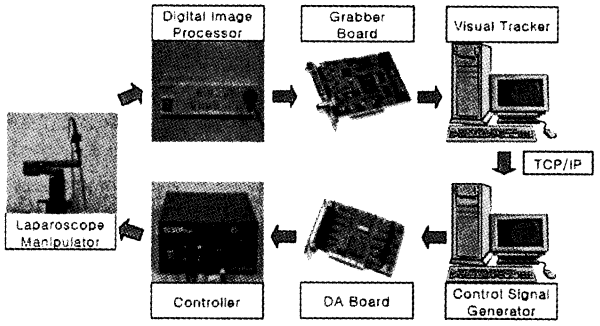


Figure 5 Experimental setup

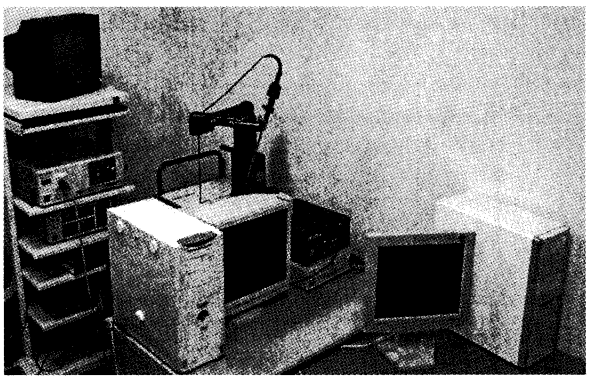


Figure 6 Experimental Equipments

We use the Olympus AR-TX model CCD, Olympus OTV-SX digital image processor and the rigid scope with the 30 degree of view range which are generally used in the laparoscope surgery. And we use the MATROX Meteor II std Grabber board and Sensoray 626 D/A board. And we apply our algorithm to the AESOP 1000 manipulator and AESOP 2000 controller.

3. In-vitro Experimental Results

3.1. Adaptability to illumination change

We tested the tracker that includes the two-stage adaptive CONDENSATION algorithm for illumination change from the normal state to the abnormal state. The images of figure 7 (a) show the good performance of the tracker in the normal state. The cross mark indicates the end position of the tip part accurately. Although the illumination changes abruptly, the cross tracker still detects the end position of the surgical tool, as can be seen in figure 7 (b), by updating the color features of the color features of the surgical tool continuously. Moreover, the images of figure 7 (c) show the good tracking ability of the tracker in the abnormal state.

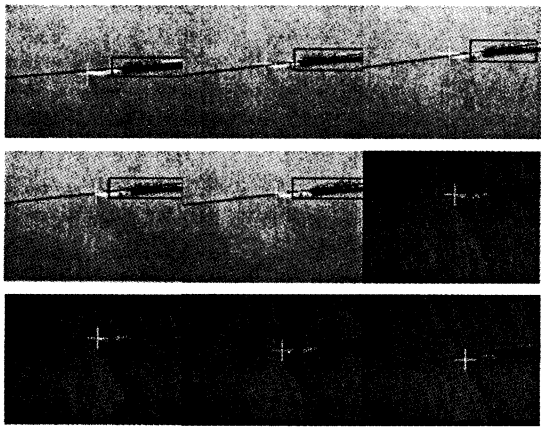


Figure 7 Experimental results (illumination changes)

Therefore, we can say that the tracker that includes the two-stage adaptive CONDENSATION algorithm is adaptable of the abrupt illumination change.

3.2 Overlapping and change of surgical tools

We tested the visual tracker in two surgical situations, intersection and change of the surgical tools, which frequently happen in real laparoscopic surgery. Figure 8 shows the result of the performance of the visual tracker under the condition of an overlapping of the surgical tools. In figure 8, the spots on the rod part of the surgical tool are the sampling positions for the first stage of the algorithm and the rectangle can be generated by the information of the width and the center line. The rectangle is the sampling region for the second stage, and the cross mark represents the end position of the surgical tool identified by the tracker. The visual tracker takes the left tool as the tracking target when the tip part of the left surgical tool is located in the rectangle. However, the tracker does not track the left tool once the left tool is out of the rectangle. This is the reason that the rectangle is constructed by the first stage's result of the tracker for the right tool. Although the tracking target is overlapped by the left surgical forceps, the tracker can detect the end position of the surgical tool.

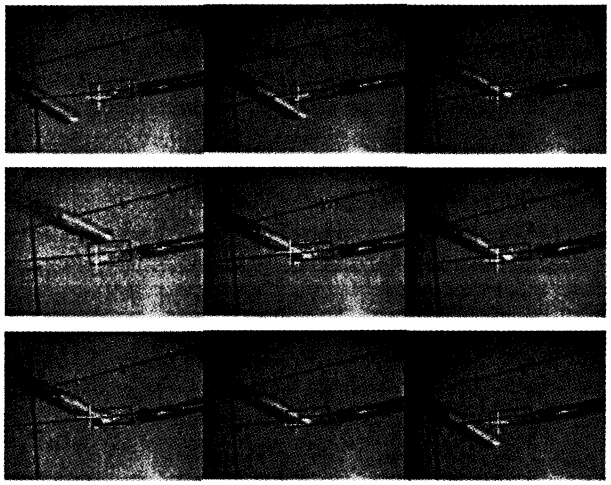


Figure 8 Experimental results (tool overlapping)

Figure 9 shows the result of a tool change. We change the left surgical tool from a pair of grasping forceps to a pair of scissors. We set a surgical situation in which a surgeon grips the grasping forceps with the right hand and he or she exchanges the grasping forceps for the laparoscopic scissors. Therefore, the tracking target changes from the grasping forceps to the scissors for the tracker. There are three rectangles in figure 9, the largest one represents the safety zone and the other rectangles represent the sampling region for each stage of the visual tracker. The sampling rectangle for the rod part simply represents the limit of the sampling positions and the sampling rectangle for the tip part is made up of the results of the first stage. The line is the center line of the surgical tool in figure 9. When the tracking target disappears in the surgical image, the tracker samples the positions in the right half of the whole image. From those images, we can see that the rectangle for the tip part disappears because that the tracker recognizes that there is no surgical tool and then set the width of the surgical tool as zero. After the visual tracker percepts a new tracking target (scissors), the tracker can detect the new tracking target continuously. From the experimental results, we can know that the tracker using the two-stage adaptive CONDENSATION algorithm is robust to the varied surgical situations and can detect the end position of surgical tools very well.

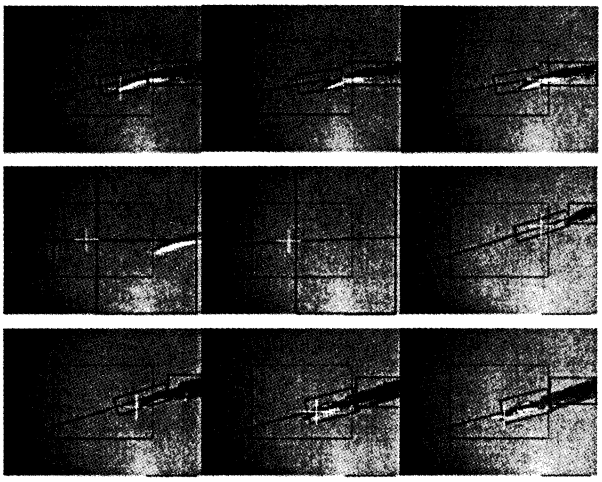


Figure 9 Experimental results (tool changes)

3.3. Real-time visual servoing

We tested the visual servoing system. In figure 10, the rectangle on the center of the surgical image represents the safety zone and the other rectangle is the sampling region for the tip part, as mentioned above. The surgical tool moves from the inside of the safety zone to the outside of the safety zone. The laparoscope manipulator locates the end position of the surgical tool inside the safety zone again. We analyzed the magnitude of the voltage signal that is the input signal of the manipulator controller versus the end position of the surgical tool in figure 11. In figure 11, XY plane is the image plane and Z axis is the magnitude of the input voltage of the controller.

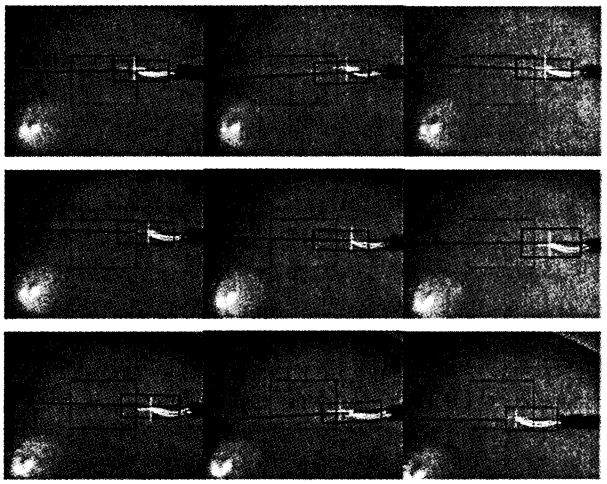


Figure 10 Experimental results of the visual servoing

As can be seen in figure 11, there is no voltage generation when the end position is located on the inside of the safety zone, otherwise voltage is generated in

proportion to the displacement between the position and the boundary of the safety zone when the end position is out of the safety zone in the image.

We also checked the processing time of the visual servoing system in the varied tests. We used 1000 sampling positions for the each stage, so the total number of sampling positions is 2000. From figure 17, we know that most of the processing times are about 40 (*ms*) and less, so the average of the frame rates is less than 25 (*frame/sec*). We expected that the processing speed of our new system may be slower than the conventional system because of the two-stage verification procedure. However, we can realize the real-time visual servoing system using the master element method at the second stage of the new algorithm.

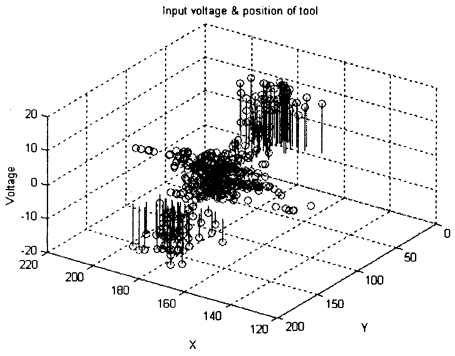


Figure 11 Input voltage of controller

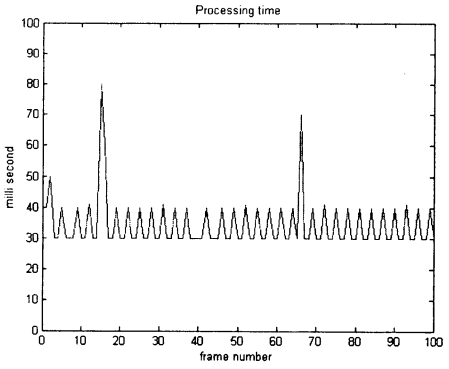


Figure 12 Image processing time

5. Conclusion

In this paper, we verify a new visual tracking algorithm, the two-stage adaptive CONDENSATION algorithm, which has real-time and robust tracking ability. The tracker, which includes the new algorithm, is adaptable to abrupt and gradual illumination changes and robust to several surgical situations (change and overlapping of laparoscopic surgical tools). We suggest a new control strategy for the laparoscopic surgery, the virtual spring algorithm. We verified the new algorithm and the new control strategy using the commercial laparoscopic robot arm AESOP 1000. Finally, we can say that our newly developed real-time visual servoing system, which includes the two-stage adaptive CONDENSATION algorithm and the virtual damper system, has sufficient performance to be a real time visual servoing system for laparoscopic surgery.

Acknowledgements

We gratefully acknowledge the financial support of the Korea Science and Engineering Foundation (HWRS-ERC).

References

[1] Maurice E. Arregui, Robert J. Fotzgibbons, Jr., Namir Katkhouida, J. Barry McKernan, Harry Reich, "Principles of Laparoscopic surgery", Springer-Verlag, 1995.

[2] Omote, K., Feussner, H., Ungeheuer, A., Arbter, K., Guo-Qing Wei, "Self-guided robotic camera control for laparoscopic surgery compared with human camera control", The American journal of surgery, Vol. 177, pp. 321-324, April, 1999.

[3] Casals, A., Amat, J., Laporte, E., "Automatic guidance of an assistant robot in laparoscopic surgery", IEEE International Conf. on Robotics and Automation, Vol. 1, pp.895–900, 1996.

[4] Cheolwhan Lee, Yuan-Fang Wang, Uecker, D.R., Yulun Wang, "Image analysis for automated tracking in robot-assisted endoscopic surgery", Proceedings of the 12th IAPR International Conference , Vol. 1, pp. 88–92, 1994.

[5] Guo-Qing Wei, Arbter, K., Hirzinger, G., "Real-time visual servoing for laparoscopic surgery. Controlling robot motion with color image segmentation", IEEE Eng. in Med. and Bio. Magazine, Vol. 16, Issue 1 , pp. 40–45, 1997.

[6] Michael Isard, Andrew Blake, "CONDENSATION-Conditional Density Propagation for Visual Tracking", Int. J. Computer Vision, 29(1), pp.5-28, 1998.

[7] Michael Isard, Andrew Blake, "Contour tracking by stochastic propagation of conditional density", In Proc. European Conf. Computer Vision, pp.343-356, 1996.

[8] David Reynard, Andrew Wilderberg, Andrew Blake, John Marchant, "Learning dynamics of complex motions from image sequences", Proc. European Conf. Computer Vision, pp. 357-368, 1996.

[9] Andrew Blake, Michael Isard, "Active contours", Springer, 1998.

[10] Rafael C. Gonzalez, Richard E. Woods, "Digital image processing", Second edition, Prentice Hall, 2001.

[11] Minseok Kim, Jin-seok Heo, Jung-Ju Lee, Real time vusual servoing for laparoscopic surgery, Proceedings of the International Symposium on Artificial Life and Robotics (AROB 8st), Beppu, Oita Japan, Feb 24-26, 2003, pp.253-256

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

An Efficient Fully Automatic Face Tracking using Binary Template Matching

Kwang Ho An
Dept. of Electrical Engineering
KAIST
Daejeon, Korea
akh@cheonji.kaist.ac.kr

Dong Hyun Yoo
Dept. of Electrical Engineering
KAIST
Daejeon, Korea
ydh@cheonji.kaist.ac.kr

Myung Jin Chung
Dept. of Electrical Engineering
KAIST
Daejeon, Korea
mjchung@ee.kaist.ac.kr

Abstract

For face tracking in a video sequence, many face tracking methods have been developed. However, most of them have a problem that the initial position of the face should be known. In this paper, we present a fast and efficient algorithm for fully automatic detection and tracking a face. Face detection is performed by binary template matching. Each sub-image extracted from the input image is binarized by an effective threshold value. The value is computed by an average intensity value. Then, the binarized sub-image is compared to predefined binary face templates. Once a face is reliably detected, we extract a skin color distribution from the detected facial region for face tracking. Tracking is realized using skin color, motion information, and binary template matching. Our proposed algorithm is computationally efficient, so that it can be executed in real-time. In addition, our algorithm is robust to face orientation, scale changes, and lighting condition changes.

Keywords: Binary template matching, Effective threshold value, Face tracking, Face detection.

1. Introduction

A face provides a variety of different communicative functions such as identification, the perception of emotional expression, and lip-reading. Many applications in human computer interaction require tracking a human face.

Robust real-time face tracking has been studied by many researchers in the past. Yang and Waibel [1] proposed a face tracking algorithm which is based on skin color filtering. However, they failed to present any concrete method for locating a face based on the results of skin color filtering. Moreover, the skin color changes when the lighting condition varies, when an different input video camera is used.

Eleftheriadis and Jacquin [2] proposed a face tracking algorithm which is based on the frame difference due to a moving head and the assumption that the outline of a person's head is elliptical. Unfortunately, there usually exists the substantial movement of the background and different objects. The ellipse fitting method may also fail when partial occlusions occur. In addition, the proposed method requires expensive computation making real-time working difficult.

In this paper, we propose a fast and efficient algorithm for fully automatic initial detection and tracking of a face. When our face tracking system starts, initial face detection is performed by binary template matching. Once a face is correctly detected, we build a skin color model from the detected facial region for fast face tracking robust to the lighting condition changes. Tracking is realized by skin color, motion information, and binary template matching.

In section 2, we describe our proposed initial face detection method. Section 3 describes how to extract a skin color model from the detected facial region and track the detected face using skin color, motion information, and binary template matching. In section 4, we show various experimental results. Section 5 concludes the paper.

2. Face Detection

Our proposed face detection system is shown in Fig. 1. We apply a window scanning technique for detecting a face. All sub-images of 20 by 20 pixels are extracted from the original input image. Each sub-image is binarized by an effective threshold value. Then, the binarized sub-image is compared to predefined binary face templates.

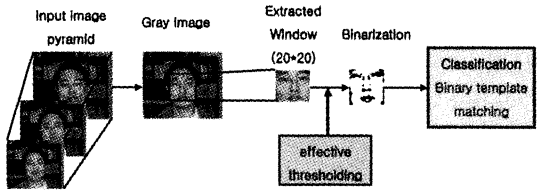


Fig. 1: Face detection system

2.1. Effective threshold value

We can expect that there may be an approximately linear relationship between the average intensity value of a sub-image and an effective threshold value. To find the approximately linear relationship, we made an experiment. The experimental set consisted of 100 hand labeled faces scaled and aligned to a base resolution of 20 by 20 pixels. They had various intensity values. Each face image was compared to the predefined binary face template shown in Fig. 2.

Fig. 2: Binary face template

For each face image, we could find both initial and final binarization threshold value which guaranteed the correlation value with the predefined binary face template to be above the decision threshold value. We define the mid-value between the two threshold values as an effective threshold value. Fig. 3 shows the relationship between the average intensity values of face images and their effective threshold values. From the Fig. 3, we can find that there is an approximately linear relationship. We can set up a linear equation which minimizes the sum of distance from each point. Using the equation (1) and (2), we can find an effective binarization threshold value.

$$\text{average\_intensity\_value} = \frac{1}{20*20}\sum_{i=1}^{20}\sum_{j=1}^{20} I(i,j) \tag{1}$$

$$\text{effective\_threshold\_value} = 0.983*\text{average\_intensity\_value} - 18.0335 \tag{2}$$

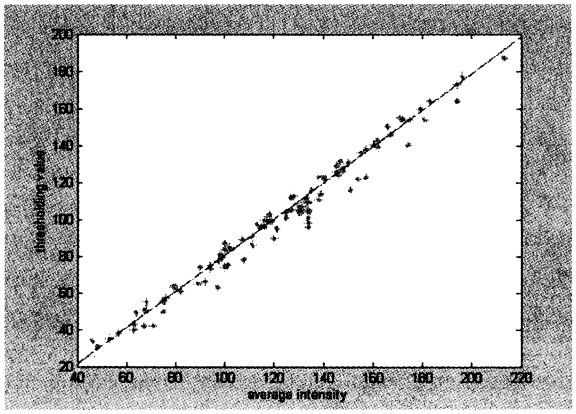


Fig. 3: Average intensity vs. threshold value

2.2. Classification

Each sub-image extracted from the input image is binarized by an effective threshold value determined from the equation (2). Then, the binarized sub-image is compared to predefined various binary face templates shown in Fig. 4. Each sub-image is classified as a face, if the computed matching value is above the predefined decision threshold value.

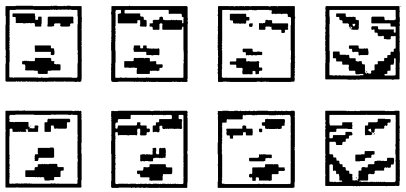


Fig. 4: Various binary face templates

3. Face Tracking

Fig. 5 shows the Block diagram of the overall face tracking system proposed in this paper. Initial face

detection is carried out by the binary template matching. Simultaneously, skin color modeling is built automatically. Then, face tracking algorithm is performed using skin color, motion information, and binary template matching.

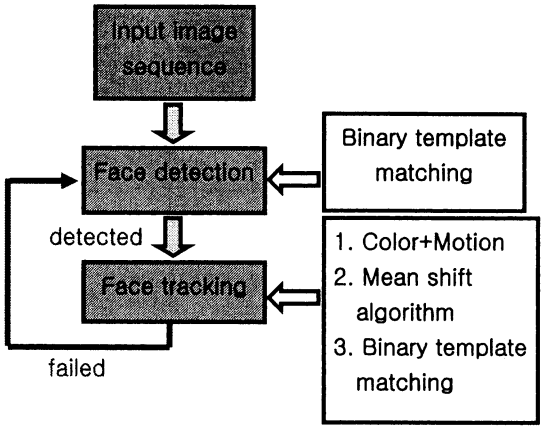


Fig. 5: Block diagram of the face tracking system

3.1. Skin color modeling

Skin color has proven to be a useful and robust cue for face detection and tracking. Color allows fast processing and is highly robust to geometric variances of the face pattern. So, we use skin color as a feature for face tracking. Since the main variation in skin appearance is largely due to luminance change, we can reduce the effect of luminance using Normalized RGB colors for skin color modeling. Skin color distribution can be modeled by an elliptical Gaussian joint probability density function $N(\mathbf{m},\Sigma)$ defined as:

$$\mathbf{m}=[\bar{r},\bar{g}]^T \tag{3}$$

$$\bar{r}=\frac{1}{n}\sum_{i=1}^{n} r_i,\quad \bar{g}=\frac{1}{n}\sum_{i=1}^{n} g_i \tag{4}$$

$$\Sigma=\begin{bmatrix}\sigma_{rr}^{\ 2} & \sigma_{rg}^{\ 2}\\ \sigma_{gr}^{\ 2} & \sigma_{gg}^{\ 2}\end{bmatrix} \tag{5}$$

$$p(\mathbf{x})=\frac{1}{2\pi|\Sigma|^{1/2}}\exp(-\frac{1}{2}[\mathbf{x}-\mathbf{m}]^T\Sigma^{-1}[\mathbf{x}-\mathbf{m}]) \tag{6}$$

Here, $\mathbf{x}=[r,g]^T$ is a color vector, and $\mathbf{m}$ and Σ are the distribution parameters (mean vector and covariance matrix respectively). The model parameters are automatically extracted from the facial region to reduce the effect of lighting condition changes and different faces when initial face detection is correctly completed by the binary template matching. The skin color filtering process classifies the pixels of a given input image into skin and non-skin pixels. A given pixel is classified into a skin pixel, if the probability in the equation (6) is greater than a predefined threshold value.

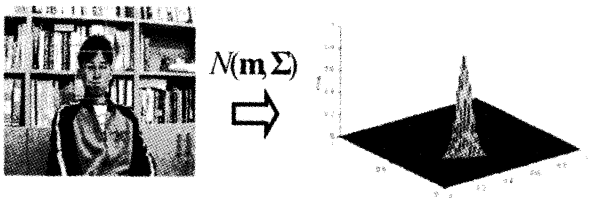


Fig. 6: Skin color modeling

3.2. Motion

Skin color alone is usually not sufficient to detect or track faces because there may exist skin colored objects in the background. Therefore, we need to use motion information for tracking a face in the complex background. Motion information is a convenient means of locating moving objects. A straightforward way to achieve motion segmentation is by frame difference analysis. Of the skin colored objects, a face can be segmented by motion filtering.

Fig. 7: Color and motion filtering

3.3. Mean shift algorithm

The mean shift algorithm is a statistically robust method for finding the peak of a probability distribution. Gary R.Bradski first used the mean shift algorithm in the face tracking [3]. We modified this mean shift tracking algorithm to use a color and motion probability distribution.

How to perform the Mean Shift Algorithm

1. Choose a search window size to contain the detected face by binary template matching.
2. Put the initial search window in the center of the detected face.
3. Compute the mean location and the size of a color and motion probability distribution in the search window.
4. Center the search window at the mean location computed in step 3 and rescale the search window size.
5. Repeat step 3 and 4 until convergence.

3.4. Binary template matching

While tracking a face, the binary template matching is carried out to detect a face correctly. Our proposed face tracking algorithm excludes moving objects which have a skin color such as hands, another distracting face using the binary template matching within the search window.

4. Experiments

Many experiments have been carried out to evaluate the proposed tracking algorithm. Our face tracking algorithm is implemented on a 1.4 GHz Pentium IV processor, providing real-time operation with 20 frames per second. Fig. 8 and 9 show various experimental results.

5. Conclusion

This paper presents a real-time face tracking algorithm which is based on skin color, motion information, and binary template matching. One unique feature of the proposed algorithm is that it automatically determines the initial face location and size using binary template matching and builds a skin color modeling from the detected facial region. Also, our proposed algorithm is computationally efficient, so that it can be executed in real-time. In addition to, the proposed algorithm is insensitive to a variety of factors including partial occlusions, face orientation, and changes in scale and lighting conditions.

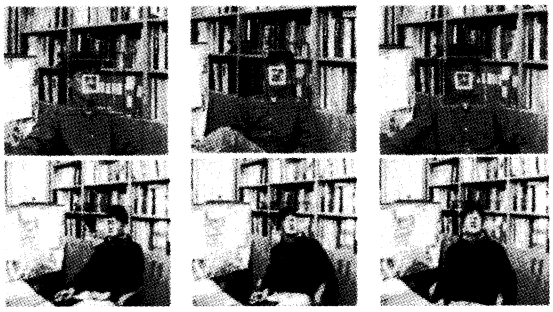


Fig. 8: Detection of a face with orientation

(a) Another face

(b) hand

Fig. 9: Tracking a face with distractors

References

[1] J. Yang and A. Waibel, "Tracking human faces in real-time," Proc. IEEE Workshop on Application of Computer Vision, 1996.

[2] A. Eleftheriadis and A. Jacquin, "Automatic face location detection and tracking for model-assisted coding of video teleconferencing sequences at low bit-rates," Signal Processing: Image Communication, No. 7, pp. 231-248, 1995.

[3] G.R. Bradski, "Real Time Face and Object Tracking as a Component of a Perceptual User Interface," Proc. IEEE Workshop on Application of Computer Vision, Princeton, 214-219, 1998.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Nonparametric method for nonlinear system identification

Hiroshi KASHIWAGI* and Yun Li†

Abstract: This paper describes a new method for identifying and separating Volterra kernels of nonlinear control systems by use of pseudorandom M-sequence and correlation technique. By use of this method, we can obtain Volterra kernels of up to 3rd order of a nonlinear system. M-sequence is applied to a nonlinear system and the crosscorrelation function between the input and output is calculated. Then the crosscorrelation function includes all the crosssections of Volterra kernels of the nonlinear system. The problem is how to separate these crosssections from each other. This paper proposes a practical method for this purpose. Computer simulations show that this method for identification and separation of Volterra kernels of nonlinear system is very effective for identifying nonlinear control systems.

Keywords: nonlinear system, identification, correlation, M-sequence, Volterra kernel

I. Introduction

Identification of nonlinear control systems is an important task for controlling a complex and nonlinear dynamical system, since almost all existing control systems are strictly speaking nonlinear. Identification methods for linear systems are investigated by many researchers and now we have a lot of strategies for identifying linear systems. However, there are not so many methods for identification of nonlinear systems. The identification methods for nonlinear systems are devided into two categories: parametric method and nonparametric method. Parametric methods include block-oriented model [10], NARX (nonlinear autoregressire exhogeneous) model [14], neural network model [4], fuzzy model [3], and frequency domain approach [5]. And nonparametric methods of nonlinear system identification include those system representation methods as Volterra kernel or Wiener kernel. Among these methods, Volterra kernel model is one of the most general models, since Volterra kernel model is considered to be an extension of impulse response model of linear case to nonlinear case. This paper is concerned with Volterra kernel identification of nonlinear systems.

Lee and Schetzen [9] proposed the use of multidimensional correlation for obtaining Wiener kernel in case where the input to the system is white-Gaussian signal. Hooper and Gyptopoulos [6] suggested the use of pseudorandom ternary signal for obtaining 2nd Volterra kernel, utilizing a property of the signal that odd order moment of the signal is zero. Barker *et al* [1] reported a method for obtaining 2nd Volterra kernel by use of pseudorandom antisymmetric M-sequence under the limited condition. Sutter [12] suggested the use of M-sequence as the stimulus for obtaining Volterra kernels of nonlinear biological system, showing that the first-order input crosscorrelation includes all the binary kernels together with an evoked potential experiment result. Shi and Hecox [13] extended this method by use of m-pulse, which is obtained from binary M-sequence, for measuring Volterra kernels of the brainstem auditory evoked response.

But both Sutter and Shi-Hecox did not show a systematic method for separating various kernel slices appearing in the crosscorrelation between the input and output. They suggested a proper selection of feedback taps of M-sequence generator, without referring to any further.

The authors have recently developed a method for obtaining Volterra kernels of up to 3rd order [7]. A pseudorandom M-sequence is applied to a nonlinear system to be identified. Then the crosscorrelation function between the input and its output is calculated. This crosscorrelation function is shown to include all crosssections of higher order Volterra kernels except when some of the arguments are equal. The problem was how to separate the significant crosssections from each other. The authors showed an example [7] of suitably selected M-sequence so as these crosssections become apart from each other.

II. The principle of identification of Volterra kernel by use of M-sequence

1. M-sequence

Before the principle of Volterra kernel identification is described, a basic general property of M-sequence is explained. M-sequence is one of the pseudorandom sequences which can be easily generated by use of simple n-stage shift register with feedback as shown in **Fig.1**. Each stage of the shift register contains 0 or 1, and each output is multiplied by a coefficient f_j (equal to 0 or 1) and added mod 2 and fed back. In the circuit, $\oplus$ denotes an exclusive OR circuit. The initial condition of the shift register can be taken arbitrarily except for all zero. When the feedback coefficients f_j are suitably chosen, the generated sequence $\{a_i\}$ has the maximum period $N = 2^n - 1$

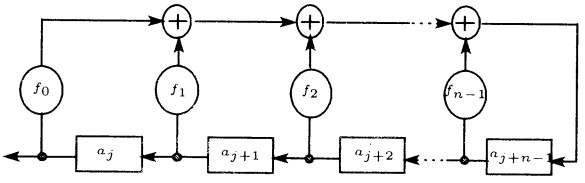


Fig.1 M-sequence generator

*Faculty of Engineering, Kumamoto University, Kurokami 2-39-1, Kumamoto, 860-8555, Japan
†Dept.Electronics adn Electrical Engineering, University of Glasgow, UK

and is called a maximum-length sequence (M-sequence). The sequence $\{a_i\}$ is written as

$$a_{i+n} = \sum_{j=0}^{n-1} f_i a_{i+j} \qquad (mod 2) \qquad (1)$$

Letting $f_n = 1$, we have

$$\sum_{j=0}^{n} f_j a_{i+j} = 0 \qquad (mod 2) \qquad (2)$$

These expressions are called linear recurrence equations. When we introduce a delay operator x such as $a_{i+n} = x^j a_i$, Eqn.(2) becomes

$$\left(\sum_{j=0}^{n} f_j x^j\right) a_i = 0 \qquad (3)$$

Here, the polynomial

$$f(x) = \sum_{j=0}^{n} f_j x^j \qquad (f_0 \neq 0, f_n = 1) \qquad (4)$$

is called the characteristic polynomial. The coefficients f_j or the sequence a_i are 0 or 1, and the multiplication and addition between them obey mod 2 arithmetic. Therefore, the coefficients f_j or sequence a_i are considered to belong to Galois Field $GF(2)$. $f(x)$ is usually denoted in octal notation by looking $\{f_j\}$ as octal number. For example, when $f(x) = x^{10} + x^3 + 1$, ($\{f_i\} = 10000001001$), $f(x) = 2011$ in octal notation.

The necessary and sufficient condition that the sequence $\{a_i\}$ is an M-sequence is that the characteristic polynomial $f(x)$ is a primitive polynomial ove $GF(2)$. Primitive polynomials over $GF(2)$ are found in Peterson [11].

When a_i ($a_i = 0$ or 1) is converted to $\{m_i\}$ ($m_i = 1$ or -1) by $m_i = 1 - 2a_i$, the autocorrelation function $\phi_{mm}(k)$ of the sequence $\{m_i\}$ is given by

$$\begin{aligned} \phi_{mm}(k) &= \frac{1}{N}\sum_{i=0}^{N-1} m_i m_{i+k} \\ &= \begin{cases} 1 & \text{for } k = 0, N, 2N, \cdots \\ -\frac{1}{N} & \text{otherwise} \end{cases} \end{aligned} \qquad (5)$$

When N is large, $\phi_{mm}(k)$ is approximately equal to δ-function, so the M-sequence becomes an almost white random signal.

When a k-shifted version of a_i is denoted by a_{i+k}, there exists an unique j (mod N) such that

$$\{a_i + a_{i+k}\} = \{a_{i+j}\} \qquad (6)$$

This property is called the shift and add property of M-sequences. In general, there exists an unique v such that

$$s_1 a_{i-1} + s_2 a_{i-2} + \cdots + s_n a_{i-n} = a_{i+v} \qquad (7)$$

for arbitrary combination of $s_1, s_2, \cdots s_n$ which belong to $GF(2)$.

Identification of Volterra kernel by use of M-sequence

The principle for obtaining Volterra kernels by use of M-sequence is here briefly reviewed [?], [7].

Consider the identification of a nonlinear system which can be described as follows,

$$y(t) = \sum_{i=1}^{\infty} \int_0^\infty \int_0^\infty \cdots \int_0^\infty g_i(\tau_1, \tau_2, \cdots \tau_i) \times u(t-\tau_1)u(t-\tau_2)\cdots u(t-\tau_i) d\tau_1 d\tau_2 \cdots d\tau_i + n(t) \qquad (8)$$

where $u(t)$ is the input, $y(t)$ is the output of the nonlinear system, $n(t)$ is noise, and $g_i(\tau_1, \tau_2, ...)$ is called Volterra kernel of i-th order. When $i = 1$, Eqn.(8) shows a linear system.

In order to get the Volterra kernels $g_i(\tau_1, \tau_2, ...)$, we use a pseudorandom M-sequence as an input to the nonlinear system. The crosscorrelation function $\phi_{uy}(\tau)$ between the input $u(t)$ and the output $y(t)$ can be written as,

$$\begin{aligned} &\phi_{uy}(\tau) = \overline{u(t-\tau)y(t)} \\ &= \sum_{i=1}^{\infty} \int_0^\infty \int_0^\infty \cdots \int_0^\infty g_i(\tau_1, \tau_2, \cdots \tau_i) \\ &\times \overline{u(t-\tau)u(t-\tau_1)\cdots u(t-\tau_i)} d\tau_1 d\tau_2 \cdots d\tau_i + \phi_{un}(\tau) \\ &= \sum_{i=1}^{\infty} \int_0^\infty \int_0^\infty \cdots \int_0^\infty g_i(d\tau_1 d\tau_2 \cdots d\tau_i) \\ &\times \phi_{uu\cdots u}(\tau, \tau_1, \cdots \tau_i) d\tau_1 d\tau_2 \cdots d\tau_i + \phi_{un}(\tau) \end{aligned} \qquad (9)$$

where $\overline{\quad\quad}$ denotes time average over $N\Delta t$ (N is the period of M-sequence and Δt is the time increment), $\phi_{un}(\tau)$ is the crosscorrelation between input $u(t)$ and noise $n(t)$, and $\phi_{uu\cdots u}(\tau, \tau_1, \cdots \tau_i)$ denotes $(i+1)$th moment of input $u(t)$. Usually the n-th moment of $u(t)$ is difficult to obtain. But when we use M-sequence, we can get n-th moment of $u(t)$ easily. Namely, when we use a two-level M-sequence as an input ($u(t) = 1$ or -1), the $(i+1)$th moment of the input M-sequence $u(t)$ can be written as

$$\begin{aligned} &\overline{u(t-\tau)u(t-\tau_1)u(t-\tau_2)\cdots u(t-\tau_i)} \\ &= \begin{cases} 1 & \text{(for certain } \tau) \\ -1/N & \text{(otherwise)} \end{cases} \end{aligned} \qquad (10)$$

When we use the M-sequence with the degree greater than 16, $1/N$ is in the order of 10^{-5}. So Eqn.(10) can be approximated as a set of impulses which appear at certain τ's.

Let us consider the case where we measure i-th Volterra kernel. Then for any integer $k_{i1}^{(j)}, k_{i2}^{(j)}, \cdots k_{i,i-1}^{(j)}$ (suppose $k_{i1}^{(j)} < k_{i2}^{(j)} < \cdots, k_{ii-1}^{(j)}$), there exists a unique $k_{ii}^{(j)} (mod\ N)$ such that

$$\begin{aligned} &u(t)u(t + k_{i1}^{(j)}\Delta t)u(t + k_{i2}^{(j)}\Delta t)\cdots u(t + k_{i,i-1}^{(j)}\Delta t) \\ = \;& u(t + k_{ii}^{(j)}\Delta t) \end{aligned} \qquad (11)$$

where j is the number of a group $(k_{i1}, k_{i2}, \cdots, k_{i,i-1})$ for which Eqn.(11) holds. This property is due to "shift and add property" (Eqn.(7)) of the M-sequence. We assume that total number of those groups is m_i(that is, $j = 1, 2, \cdots, m_i$). Therefore Eqn.(10) becomes unity when

$$\tau_1 = \tau - k_{i1}^{(j)}\Delta t, \tau_2 = \tau - k_{i2}^{(j)}\Delta t, \cdots \tau_i = \tau - k_{ii}^{(j)}\Delta t \qquad (12)$$

Therefore Eqn.(9) becomes

$$\begin{aligned}\phi_{uy}(1;\tau) &\simeq \sum_{i=1}^{\infty}\sum_{j=1}^{m_i} g_i(\tau - k_{i1}^{(j)}\Delta t, \tau - k_{i2}^{(j)}\Delta t, \\ &\qquad \cdots \tau - k_{ii}^{(j)}\Delta t) + \phi_{un}(\tau) \\ &\simeq \Delta t g_1(\tau) + F(\tau) \\ &\quad + \sum_{i=2}^{\infty} i!(\Delta t)^i \sum_{j=1}^{m_i} g_i(\tau - k_{i1}^{(j)}\Delta t, \tau - k_{i2}^{(j)}\Delta t, \\ &\qquad \cdots, \tau - k_{ii}^{(j)}\Delta t) + \phi_{un}(\tau)\end{aligned} \tag{13}$$

where $\phi_{uy}(1;\tau)$ denotes the crosscorrelation function between $u(t)$ and $y(t)$ when the amplitude of $u(t)$ is 1. Δt is the time increment. Here the function $F(\tau)$ is the function of τ and sum of the odd order Volterra kernels when some of its arguments are equal. Since $F(\tau)$ appears together with $g_1(\tau)$ in a overlapped manner, $F(\tau)$ must be calculated from the measured odd order Volterra kernels and be subtracted from the measured $\phi_{uy}(1;\tau)$ in order to obtain the accurate $g_1(\tau)$. Since $g_i(\tau_1, \tau_2, \cdots \tau_i)$ is zero when any of τ_i's are smaller than zero, each $g_i(\tau - k_{i1}^{(j)}\Delta t, \tau - k_{i2}^{(j)}\Delta t, \cdots \tau - k_{ii}^{(j)}\Delta t)$ in Eqn.(8) appear in the crosscorrelation function $\phi_{uy}(1;\tau)$ when $\tau > k_{ii}^{(j)}\Delta t$. $g_i(\tau - k_{i1}^{(j)}\Delta t, \tau - k_{i2}^{(j)}\Delta t, \cdots \tau - k_{ii}^{(j)}\Delta t)$ is a crosssection of i-th Volterra kernel sliced in the diagonal direction. For example, **Fig.2** shows an example for a second order Volterra kernel, and $g_2(\tau - k_{21}\Delta t, \tau - k_{22}\Delta t)$ becomes a crosssection sliced along the direction of 45^o line in $\tau_1 - \tau_2$ plane, starting at $\tau_1 = 0$, $\tau_2 = k_{22}\Delta t - k_{21}\Delta t$. So this crosssection $g_2(\tau - k_{21}\Delta t, \tau - k_{22}\Delta t)$ is called "kernel slice" hereafter.

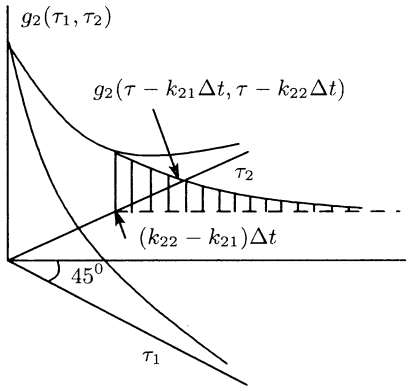


Fig.2 Example of second order Volterra kernel

Eqn.(13) means that, unlike the multidimensionel correlation method by Lee and Schetzen, the one dimensional crosscorrelation $\phi_{uy}(\tau)$ between the input $u(t)$ and output $y(t)$ contains all the kernel slices of Volterra kernels of order less than or equal to n, since from Eqn.(11) there exist unique $k_{ii}^{(j)}$'s for any i less than or equal to n. Among these kernel slices, we would like to obtain significant, or prominent slices for characterizing the dynamic characteristics of the nonlinear systerm. Since Volterra kernels become small when some of the argument become large, we concentrate to measure kernel slices in which all the time variables τ_i's are less than, say, $20\Delta t$.

In order to obtain the Volterra kernels from Eqn.(13), $k_{ii}^{(j)}$'s must be sufficiently apart from each other. For this to be realized, we have to select suitable M-sequences, which make the appearance of each kernel slice of Volterra kernels sufficiently apart each other.

III. Simulation

This method is applied to a process problem of Mitsubishi Chemical Corp. The chemical reactor is described by the differential equation

$$\begin{aligned}\frac{dx_1}{dt} &= \frac{1}{Tp_1}(-x_1 + Kp_1 u_1) \\ \frac{dx_2}{dt} &= \frac{1}{Tp_2}(Kp_2 x_1 x_2 - x_2 + Kp_3 u_2) \\ y &= x_2\end{aligned} \tag{14}$$

where

$$\begin{aligned}u_1 &= 0.05(kg/h), \quad u_2 = 3195(kg/h) \\ Kp_1 &= 0.4, \qquad Kp_2 = -1648(1/kg/h) \\ Kp_3 &= 0.05317(kg/cm^2/kg/h) \\ Tp_1 &= 2.4(h), \qquad Tp_2 = 7.1(h) \\ x_1 &= 0.02(kg/h), \quad x_2 = 5.0(kg/cm^2)\end{aligned}$$

and x_1 is the consumption velocity of catalyst, x_2 is gas density, u_1 is the supply quantity of catalyst, u_2 is the supply quantity of polyethylene, and Tp_1, Tp_2, Kp_1, Kp_2, Kp_3 are constant. Here the input $u(t)$ to this process is thought to be u_1 and the output $y(t)$ is x_2.

For this, we apply an M-sequence of amplitude of ± 0.025, which is denoted as Δu, with a characteristic polynomial $f(x) = 260577$ in octal notation is applied to the nonlinear process, with a sampling period of 0.3 hour. Taking the cross-correlation between Δu and Δy, Volterra kernels are obtained. These are shown in **Fig.3**, **Fig.4** and **Fig.5**, respectively. In order to validate the measured Volterra kernels, comparison is made between the actual output and the output reconstructed from the measured Volterra kernels responding to a sinusoidal input, **Fig.6** details the comparison for the cases of 3rd Volterra kernels. We see that the third-order Volterra model reflects well about the nonlinear system.

VI. Conclusion

In this paper, a new method for identifying and separating Volterra kernels of nonlinear control system is described. With suitable selection of M-sequence, we can obtain up to the third order Volterra kernels of nonlinear systems.

Simulation results show the effectiveness of the methods indicating wide applications of this method for identification of nonlinear systems.

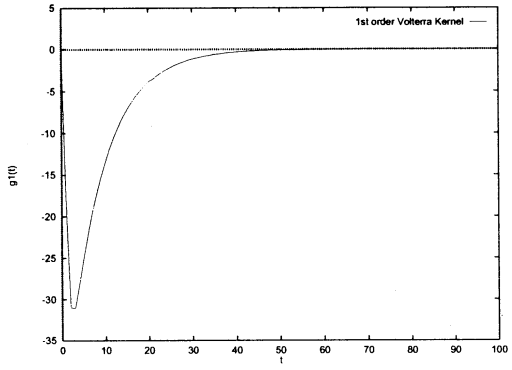


Fig.3 Obtained first-order Volterra kernel of the gaseous phase concentration process

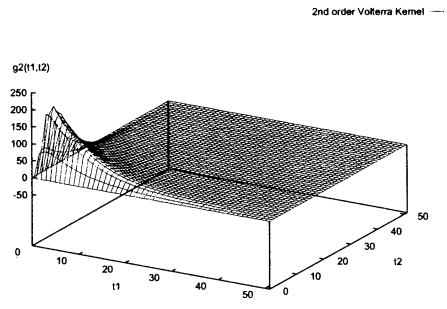


Fig.4 Obtained second-order Volterra kernel of the gaseous phase concentration process

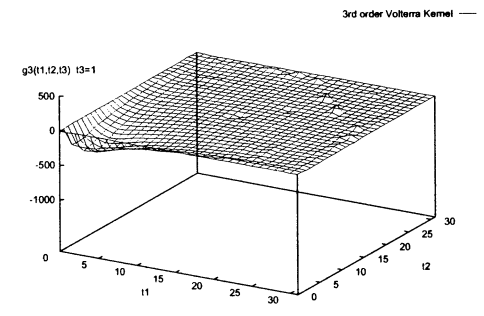


Fig.5 Obtained third-order Volterra kernel of the gaseous phase concentration process

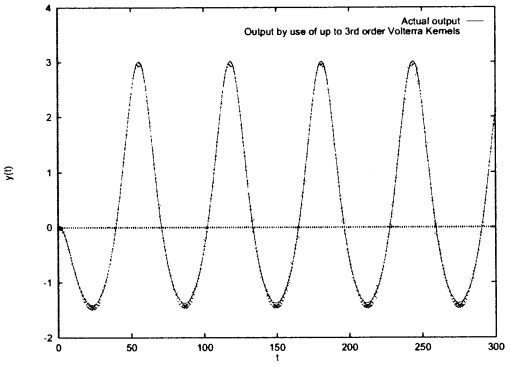


Fig.6 Comparison between the actual output and the output reconstructed from a third-order Volterra model of the gaseous phase concentration process

References

[1] H.A. Barker and S.N. Obidegwo, "Combined cross-correlation method for measurement of 2nd-ordr Vol-terra kernels," *ibid*, Vol.120, no.1 pp.114-118, 1973

[2] S.A. Billings, "Identification of nonlinear systems-a survey," Proc. IEE, Vol.127, Pt.D, No.6 pp.272-285, 1980

[3] R.Babuška and H.B.Verbruggen, "An overview of fuzzy modeling for control," Control Engineering Practice, Vol.4, No.11, pp.1593-1606, 1996

[4] S. Chen, S.A.Billings and P.M. Grant, "Non-linear system identification using neural network," Int. J. Control, Vol.51, No.6, pp.1191-1214, 1990

[5] A.S. French and E.G. Butz, "Measuring the Wiener kernels of a nonlinear system using the FFT," Int. J. Control, Vol.17, pp.529-539, 1973

[6] R.J. Hooper and E.P. Gyftopoulos, "On the measurement of characteristic kernels of a class of nonlinear systems," Proc. Symp. on Neutron Noise, Waves and Pulse Propagation, Univ. of Florida, Gainesville, pp.343-353, 1966

[7] H. Kashiwagi and Sun Yeping, "A method for identifying Volterra kernels of nonlinear systems," Trans. SICE, Vol.31, No.8, pp.1054-1060, 1995

[8] H. Kashiwagi, "M-sequence and its applications, " Shoukoudo co., Japan, 1996

[9] Y.W. Lee and M. Schetzen, "Measurement of the Wiener kernels of a nonlinear system by crosscorrelation," Int. J. Control, Vol.2 pp.237-254, 1965

[10] R.K. Pearson, "Nonlinear input / output modeling," Journal of Process Control, Vol.5, No.4 pp.197-211, 1995

[11] W.W.Peterson, "Error-Correcting Codes,", MIT Press, 1961

[12] E.E. Sutter, "A practical nonstochastic approach to nonlinear time-domain analysis," in Advanced Methods of Physiological System Modeling, V.Z. Marmarelis, Ed. Los Angeles, Univ. Southern California, 1987

[13] Y. Shi and K.E. Hecox, "Nonlinear system identification by m-pulse sequences: Application to brainstem auditory evoked responses," Trans. IEEE on Viomed. Eng., Vol.38, No.9 pp.834-845, 1991

[14] X.Zhao and V.Z.Marmarelis, "Nonlinear parametric models from Volterra kernels measurements," Mathl. comput. Modelling, Vol.27, No.5 pp.37-43, 1998

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Modelling Artificial Life by Attributed Eco-Array Grammars

Rudolf FREUND Marion OSWALD Aneta BINDER
Faculty of Informatics
Vienna University of Technology
Vienna, Austria
rudi@emcc.at marion@emcc.at ani@emcc.at

Abstract

For modelling agents and their actions as well as their interactions with the environment we propose eco-array grammars (multi-level n-dimensional parallel array grammars). Even complex system behaviour can be captured by adding attributes to this syntactic description model.

1 Introduction

In the area of formal language theory, eco grammar systems have become an interesting field for the syntactic description of the interaction between agents within a dynamically changing environment. In that way, formal syntactical models can be used for describing the evolution of complex systems in the area of artificial life. Modelling agents and their environment in a two- or three- dimensional space can be accomplished by using eco-array grammars, which are parallel variants of multi-level n-dimensional array grammars (as introduced in [2]). In eco-array grammars, rules performing the microscopic interactions between components of an agent find their formal syntactic implementation. From these microscopic reactions the more complex behaviour at the macroscopic level can be modelled, too. Even interactions with the environment can be captured at the molecular level, and by adding more complex rules at higher levels, not only reflex actions but also movements and reactions based on higher brain activity can be introduced into this model. The theoretical foundation and the formal language theory background is based on the idea of Conway's famous game of life (e.g., see [4]). The theoretical model we propose for modelling actions of agents and their interaction with their environment allows us to describe the status of the objects under consideration at different levels of scaling, from their atomic reactions to their overall behaviour. The dynamic processes within the objects and between different objects are accomplished by the application of parallel rules at each level of the eco-array grammars. To implement a more complex system behaviour, we use attributes at specific levels of object description which allows for including specific informations as well as for complex interactions. Moreover, using attributes we may even implement neural networks at some specific level of the attributed eco-array grammar. In that way, complex system behaviour can be learnt from examples and dynamic system evolution becomes possible.

2 Preliminaries

In this section we introduce the definitions and notations for arrays as well as for (parallel) array grammars; for more details, notions and results from formal language theory the reader is referred to [7].

Let Z denote the set of integers, let N denote the set of positive integers, i. e. $N = \{1, 2, ...\}$, and let $n \in N$. The norm $\|u\|$ of a vector $u \in Z^n$, $u = (u(1), ..., u(n))$, is defined by $\|u\| = \max\{|u(i)| \mid 1 \leq i \leq n\}$.

An *n-dimensional array* $\mathcal{A}$ over an alphabet V is a function $\mathcal{A} : Z^n \rightarrow V \cup \{\#\}$ with finite support *supp*$(\mathcal{A})$, where $supp(\mathcal{A}) = \{v \in Z^n \mid \mathcal{A}(v) \neq \#\}$; $\# \notin V$ is called the *background* or *blank symbol*. We usually shall write $\mathcal{A} = \{(v, \mathcal{A}(v)) \mid v \in supp(\mathcal{A})\}$.

The set of all n-dimensional arrays over V shall be denoted by V^{*n}. The *empty array* in V^{*n} with empty support shall be denoted by Λ_n.

Let $v \in Z^n$. Then the *translation* $\tau_v : Z^n \rightarrow Z^n$ is defined by $\tau_v(w) = w + v$ for all $w \in Z^n$, and for any array $\mathcal{A} \in V^{*n}$ we define $\tau_v(\mathcal{A})$, the corresponding n-dimensional array translated by v, by $(\tau_v(\mathcal{A}))(w) = \mathcal{A}(w - v)$ for all $w \in Z^n$. The vector $(0, ..., 0) \in Z^n$ shall be denoted by Ω_n.

3 N-dimensional Lindenmayer systems

N-dimensional Lindenmayer systems are parallel n-dimensional array grammars with parallel applications of interactive rules.

A parallel array production over an alphabet V is a triple $(X, \{(v, X_v) \mid v \in U\}, Y)$, where U is a finite subset of $Z^n - \{\Omega_n\}$ and X, Y, andX_v, for all $v \in U$, are elements of V. A finite set T of parallel array productions over the alphabet V is called a *table of parallel array productions*. An array $\mathcal{B}_2 \in V^{*n}$ is said to be *directly derivable* from the array $\mathcal{B}_2 \in V^{*n}$ by T, if and only if for each $w \in Z^n$ a parallel array production $(X, \{(v, X_v) \mid v \in U\}, Y)$ exists such that $\mathcal{B}_1(w) = X$, $\mathcal{B}_2(w) = Y$, and $\mathcal{B}_1(v + w) = X_v$ for all $v \in U$, and we write $\mathcal{B}_1 \Longrightarrow_T \mathcal{B}_2$.

Moreover we shall make the convention that the blank symbol # can only be replaced in the non-empty context by a symbol of V, i. e. $Y \neq \#$ is allowed for $X = \#$ only if $X_v \neq \#$ for some $v \in U$. As a consequence of this convention changes in an n-dimensional array by using a table of parallel productions are restricted to a finite area of the space Z^n.

An *n-dimensional Lindenmayer system* is a quintuple $S = (n, V, \Sigma, P, \#)$, where $n \in N$ is the dimension of the system, $V - \Sigma$ is a set of non-terminal symbols, Σ is a set of terminal symbols, and P is a finite set of tables of parallel array productions over V.

An array $\mathcal{B}_2 \in V^{*n}$ is said to be *directly derivable* from the array $\mathcal{B}_1 \in V^{*n}$ in S, if and only if there is a table T of parallel productions in P such that $\mathcal{B}_1 \Longrightarrow_T \mathcal{B}_2$, and we write $\mathcal{B}_1 \Longrightarrow_S \mathcal{B}_2$; $\Longrightarrow_S^*$ denotes the reflexive and transitive closure of the relation $\Longrightarrow_S$.

Let $S = (n, V, \Sigma, P, \#)$ be an n-dimensional Lindenmayer system and $\mathcal{A} \in V^{*n}$. Then $G = (S, \mathcal{A})$ is called an *n-dimensional Lindenmayer grammar.*

Example 1. *Conway's game of life* ([4]) can be described by the two-dimensional Lindenmayer system $S = (2, \{1\}, \{1\}, P, \#)$ with $P = P_1 \cup P_2 \cup P_3 \cup P_4$ and

$$
\begin{aligned}
P_1 &= \{(1, \{(v, X(v)) \mid v \in U\}, 1) \mid c \in \{2, 3\}\}, \\
P_2 &= \{(1, \{(v, X(v)) \mid v \in U\}, \#) \mid c \notin \{2, 3\}\}, \\
P_3 &= \{(\#, \{(v, X(v)) \mid v \in U\}, \#) \mid c \neq 3\}, \\
P_4 &= \{(\#, \{(v, X(v)) \mid v \in U\}, 1) \mid c = 3\},
\end{aligned}
$$

where $U = \{v \in Z^2 \mid \|v\| = 1\}$ is the environment and $c = card(\{v \in U \mid X(v) = 1\})$.

In Figure 1 four successive derivation steps starting from the axiom $\mathcal{A}$ with $supp(\mathcal{A}) = \{(0,0), (1,0), (2,0), (2,1), (1,2)\}$ are depicted showing the periodicity of the evolving structures.

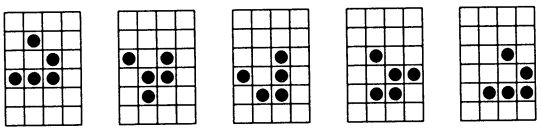


Figure 1: The "Glider".

The example given above indicates how cellular automata (which are typical representatives for an artificial life model) can be described by means of n-dimensional Lindenmayer systems. In fact, an n-dimensional *cellular automaton* can be seen as a specific type of an n-dimensional Lindenmayer system $S = (n, V, \Sigma, P, \#)$ meeting the following conditions:

1. There exists only one uniform neighbourhood in the parallel array productions in P, i. e. there exists a $U \subseteq Z^n$ such that every parallel array production in P is of the form $(X, \{(v, X_v) \mid v \in U\}, Y)$.
2. The set of parallel array productions P is deterministic in the sense that if $(X, \{(v, X_v) \mid v \in U\}, Y)$ and $(X, \{(v, X'_v) \mid v \in U\}, Y')$ are two different X-productions, then $\{(v, X_v) \mid v \in U\} \neq \{(v, X'_v) \mid v \in U\}$.

Obviously, the two-dimensional Lindenmayer system representing Conway's game of life is a (two-dimensional) cellular automaton according to this definition.

4 Attributed Eco-Array Grammars

In order to represent more-dimensional objects at different levels of scaling, we introduce the following notion:

Let $n = (n_0, n_1, ..., n_d)$, where $n_0 \in N$, $d \in N \cup \{0\}$, and $n_i = (n_{i,1}, ..., n_{i,d})$ $1 \leq i \leq d$, with $n_{i,j} \in N$, $1 \leq j \leq d$.

For each i with $1 \leq i \leq d$ we define the set of n_0-dimensional vectors

$$D_i = \{(x_1, ..., x_{n_0}) \mid 0 \leq x_k < n_{i,k},\ 1 \leq k \leq n_0\}.$$

Moreover, let $V = (V_0, V_1, ..., V_d)$, where V_i, $0 \leq i \leq d$, are (finite) alphabets, and let # be the blank symbol.

Then a (finite) *d-level array* (of dimension n) over V is an object of the following form:
$\mathcal{A} = \mathcal{A}[] = \{(v_0, \mathcal{A}(v_0), \mathcal{A}[v_0]) \mid v_0 \in supp(\mathcal{A})\}$, where $\mathcal{A}(v_0)$ is a symbol from V_0 for all $v_0 \in supp(\mathcal{A})$;

$\vdots$

$\mathcal{A}[v_0, ..., v_k] = \{(v_{k+1}, \mathcal{A}(v_0, ..., v_k, v_{k+1}),$
$\mathcal{A}[v_0, ..., v_k, v_{k+1}]) \mid v_{k+1} \in supp(\mathcal{A}[v_0, ..., v_k])\}$,
where $\mathcal{A}(v_0, ..., v_k, v_{k+1}) \in V_{k+1} \cup \{\#\}$
for all $v_{k+1} \in D_{k+1}$ and $supp(\mathcal{A}[v_0, ..., v_k]) =$
$\{v_{k+1} \in D_{k+1} \mid \mathcal{A}(v_0, ..., v_k, v_{k+1}) \neq \#\}$;
moreover, we demand
$\mathcal{A}(v_0, ..., v_k) = \# \Rightarrow supp(\mathcal{A}[v_0, ..., v_k]) = \emptyset$.

$\vdots$

$\mathcal{A}[v_0, ..., v_{d-1}] = \{(v_d, \mathcal{A}(v_0, ..., v_{d-1}, v_d)) \mid$
$v_d \in supp(\mathcal{A}[v_0, ..., v_{d-1}])\}$, where
$\mathcal{A}(v_0, ..., v_{d-1}, v_d) \in V_d \cup \{\#\}$ for all $v_d \in D_d$ and
$supp(\mathcal{A}[v_0, ..., v_{d-1}]) = \{v_d \in D_d \mid$
$\mathcal{A}(v_0, ..., v_{d-1}, v_d) \neq \#\}$; moreover we demand
$\mathcal{A}(v_0, ..., v_{d-1}) = \# \Rightarrow supp(\mathcal{A}[v_0, ..., v_{d-1}]) = \emptyset$.

The set of all these d-level arrays of dimension n over V is denoted by V^{*n}. As can be seen from the definition of a d-level array $\mathcal{A}$ in V^{*n}, the substructure of $\mathcal{A}$ at any level k, $1 \leq k \leq d$, can also be interpreted as n_0-dimensional array over V_k, where the position of the symbol $\mathcal{A}(v_0, ..., v_k)$ is given by $(x_1, ..., x_{n_0})$ and
$x_i = (n_{i,k}(n_{i,k-1}...(n_{i,1}v_0 + v_1)... + v_{k-1}) + v_k)$,
$1 \leq i \leq n_0$.

At each level, we now assign an attribute c from a given set of atrributes C to every non-blank position:
$\mathcal{A}[v_0, ..., v_k] = \{(v_{k+1}, \mathcal{A}(v_0, ..., v_k, v_{k+1}), c,$
$\mathcal{A}[v_0, ..., v_k, v_{k+1}]) \mid v_{k+1} \in supp(\mathcal{A}[v_0, ..., v_k])\}$.

In that way we obtain attributed d-level arrays of dimension n over (V, C); the set of all these attributed d-level arrays of dimension n over (V, C) is denoted by $(V, C)^{*n}$.

The model of attributed eco-array grammars defined in the following has been chosen according to the concepts of co-operation and communication: In a derivation step with a (d-level) attributed eco-array grammar, at each level a suitable derivation must be possible; the production sets at all levels co-operate in performing a derivation step; from the higher level to the lower level, the sets of productions allowed to be applied at the lower level are communicated from each cell at the higher level to every assigned cell at the lower level.

An *attributed (d-level) eco-array grammar* is a 7-tuple $G = (n, V, C, \Sigma, P, \mathcal{A}, \#)$, where d, n as well as V and C are as above, $\Sigma = (\Sigma_0, ..., \Sigma_d)$, with $\Sigma_i \subseteq V_i$, $0 \leq i \leq d$, $\mathcal{A} \in (V, C)^{*n}$ is the axiom and P is a set of tables, where each table in P is of the form $(P_0, ..., P_d)$, P_i containing (labelled) *attributed eco-array productions* of the form

$$l : (Y_0, \{(v_j, Y_j) \mid 1 \leq j \leq k\}, Y, f, L),$$

where l is the label of this production at level i, $k \geq 0$, $v_j \in Z^{n_0} - \{\Omega_{n_0}\}$, $Y, Y_j \in V_i \cup \{\#\}$, $0 \leq j \leq k$, and f is a (partial recursive) function computing a new attribute value for the position under consideration from the attribute c_0 currently assigned to this position as well as from the attributes c_j assigned to the relative positions v_j; moreover, $L \subseteq Lab_i$, where Lab_i, $0 \leq i \leq d$, is the set of labels at level i and each label is uniquely assigned to one of the productions in a set P_i.

In some sense, the attributed eco-array production $(Y_0, \{(v_j, Y_j) \mid 1 \leq j \leq k\}, Y, f, L)$ has a rewriting function like the corresponding parallel array production $(Y_0, \{(v_j, Y_j) \mid 1 \leq j \leq k\}, Y)$, but also computes a new attribute value by the function f and, moreover, specifies which table of attributed eco-array productions has to be used at the next level.

For $\mathcal{B}_1, \mathcal{B}_2 \in (V, C)^{*n}$ we now define the derivation relation $\mathcal{B}_1 \Longrightarrow_G \mathcal{B}_2$ in the following way:

1. Choose a table $(P_0, ..., P_d)$ from P.

2. Beginning at level 0, apply productions from P_0 in parallel to every position in $\mathcal{B}_1$. The sets L determine which rules from P_1 can be used at level 1.

3. Successively the derivation proceeds to level 1, ... until finally we reach level d (here we have no sets L any more). Moreover, if at some level $i < d$ the set L is empty, then we need not apply any rules at the following labels.

4. At each level i, $0 \leq i \leq d$, the parallel application of the attributed eco-array productions means the following: For every position in the n_0-dimensional array forming the substructure at level i, obtained from $\mathcal{B}_1$ by the changes at levels $0, ..., i-1$, we choose an attributed eco-array production from P_i, the label of which is consistent with the set $L \subseteq Lab_i$, determined by the production applied at level $i-1$. These productions are applied in parallel like the corresponding parallel array productions and also compute the new attribute value (observe that at every level only a finite area has to be considered, because a blank cell at level 0 can only be changed to a

non-blank cell in a non-blank environment and at every level i, a non-blank cell cannot have non-blank substructures).

Example 2. Instead of a (big) "glider" like that depicted in Figure 1 we consider a very small "boat" occupying only two neighbouring diagonal positions in the plane Z^2, which can move along this diagonal with different speeds. Such a boat with current speed s at the positions v can be described by the attributed 2-level array

$$\{(v,(x,s),\{((0,0),(a,s))\}),\\ (v+(1,1),(\bar{x},s),\{((0,0),(a,s))\})\}.$$

The first sublevel symbolizes the coherence forces that hold the boat together. For these purposes we only need one cell at level 1; hence we take $n = (2,(1,1))$, i.e. $n_{1,1} = n_{1,2} = 1$ and $D_1 = \{(0,0)\}$, and consider attributed 1-level arrays in $((V_0,V_1),C)^{*(2,(1,1))}$, where $V_0 = \{x,\bar{x},y,\bar{y}\}$ and $V_1 = \{a\}$ as well as $C = \{s \mid 2 \leq s \leq \max\}$ (max denotes the maximal speed) is the set of attributes.

The productions describing the movements of the boat are collected in P_0 for the ground level 0 and in P_1 for the first sublevel 1; the function Φ specifies the function which maps every speed s to the empty attribute (as a blank cell cannot have an attribute), $f_{1,t}$ from the empty attribute computes the new attribute t, $f_{2,t}$ from the empty attribute and an arbitrary attribute s computes the new attribute t, f_c compares two arguments and only in the case of equality yields this value as the result, and id denotes the identity function.

$$\begin{aligned} P_0 = \; & \{[x,\#]:(x,\#,\Phi,\varnothing),[\bar{x},\#]:(\bar{x},\#,\Phi,\varnothing)\}\cup\\ & \{[y,\#]:(y,\#,\Phi,\varnothing),[\bar{y},\#]:(\bar{y},\#,\Phi,\varnothing)\}\cup\\ & \{[x,t]:(\#,\{((-s,-s),x)\},y,f_{2,t},\{[y,t]\}),\\ & \quad [\bar{x},t]:(\#,\{((-s,-s),\bar{x})\},y,f_{2,t},\{[\bar{y},t]\})\\ & \quad \mid t\in C\}\cup\\ & \{[y,0]:(y,x,id,\{[y,1]\})\}\cup\\ & \{[\bar{y},0]:(\bar{y},\bar{x},id,\{[\bar{y},1]\})\}\cup \end{aligned}$$

and

$$\begin{aligned} P_1 = \; & \{[y,t]:(\#,a,f_{1,t}),[\bar{y},t]:(\#,a,f_{1,t}) \mid t\in C\}\cup\\ & \{[y,1]:(a,\{((1,1),a)\},a,f_c)\}\cup\\ & \{[\bar{y},1]:(a,\{((-1,-1),a)\},a),f_c\}. \end{aligned}$$

By these rules, from the attributed 2-level array

$$\{(v,(x,s),\{((0,0),(a,s))\}),\\ (v+(1,1),(\bar{x},s),\{((0,0),(a,s))\})\}.$$

we obtain the attributed 2-level array

$$\{(v+(s,s),(y,t),\{((0,0),(a,t))\}),\\ (v+(s+1,s+1),(\bar{y},t'),\{((0,0),(a,t'))\})\}$$

by using the rules $[x,\#]$ and $[\bar{x},\#]$ as well as $[x,t]$ and $[\bar{x},t']$ for some $t,t' \in C$ at level 0 and the rules $[y,t]$ and $[\bar{y},t']$ at level 1. Only if $t = t'$, in the next step we are able to continue the derivation succesfully by using the rules $[y,0]$ and $[\bar{y},0]$ at level 0 as well as the rules $[y,1]$ and $[\bar{y},1]$ at level 1, which allow for checking $t = t'$; for $t = t'$, we obtain the new attributed 2-level array

$$\{(v+(s,s),(x,t),\{((0,0),(a,t))\}),\\ (v+(s+1,s+1),(\bar{x},t),\{((0,0),(a,t))\})\}.$$

Using suitable attributes even the dynamic features of neural networks can be modelled, e.g., due to the underlying grid structure, Kohonen's model of self-organizing feature maps (see [5]) is especially well suited for being simulated by attributed eco-array gramars. How a specific model of (n-dimensional) attributed parallel array grammars can be used as a specification language for various models of neural networks and as a formal tool for proving specific characteristic features of these networks was already shown in [3].

References

[1] E. Csuhaj-Varjú, J. Kelemen, A. Kelemenova, Gh. Păun, *Eco(grammar)systems: A preview*, Cybernetics and Systems '94, vol. **1** (R. Trappl, ed.), World Sci. Publ., Singapore (1994), pp. 941–949

[2] R. Freund, *Multi-level eco-array systems,* in: [6], pp. 175–201

[3] R. Freund, F. Tafill, *Modeling Kohonen networks by attributed parallel array systems,* SPIE Proceedings Vol. 2093 (1994), pp. 161–172

[4] M. Gardner, The fantastic combinations of John Conway's new solitaire game "life", *Mathematical Games, Scientific American* **223** (1970), pp. 120–123

[5] T. Kohonen, *Self-Organization and Associative Memory*, Springer Series in Information Sciences **8**, Heidelberg (1984)

[6] Gh. Păun (ed.), *Artificial life: Grammatical Models*, Black Sea University Press, Romania (1995)

[7] G. Rozenberg, A. Salomaa (eds.), *Handbook of Formal Languages*, Springer-Verlag, Berlin (1997)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Multilayered Discourse in Hyper-comic

Yasuhiro Endo
Graduate School of Engineering
Yamanashi University
4-3-11 Takeda, Kofu, Yamanashi 400-8511, Japan
g02mk005@ccn.yamanashi.ac.jp

Takashi Ogata
Department of Medicine and Engineering
Yamanashi University
4-3-11 Takeda, Kofu, Yamanashi 400-8511, Japan
ogata@esi.yamanashi.ac.jp

Abstract

The theme of this paper is one part of the Hyper-comic system that we have proposed as a new form of Manga on computer, and explain the concept and mechanism of "multilayered discourse". First, we show the framework of the hyper-comic and the implementation mechanisms. Second, based on the analysis of a real Manga work, we abstract unique and interesting discourse techniques in Manga, and mainly explain techniques in multilayered discourse. Last, we show the implementation mechanism of the multilayered discourse. This discourse technique means a kind of synthesis of various rhetorical techniques, and they are naturally expressed in Manga than other narrative media.

1. Introduction

We have proposed the concept and mechanism of "Hyper-comic" system [1] [2]. It is a Manga-version of hypertext novel on computer. This research is a sub-project of "expanded literary theory" [3]. Its goal is integrating cognitive/computational approaches (like cognitive science and artificial intelligence) and literary theories to develop narrative generation systems and the application systems. Hyper-comic is a network structure applied techniques of "narrative discourse theory" proposed by Genette [4], and is our new systematization [1] to design various links among panels. Narrative discourse theory is a research area in literary theories on how to narrate a story. In the first version of hyper-comic, we developed a basic mechanism using existing discourse techniques of novels, and now, we are doing an analysis of original and unique discourse techniques in Manga to expand Genette's framework of narrative discourse. Especially, we think that one of the most unique and high level narrative techniques is "multilayered discourse", in which some different discourse techniques are expressed simultaneously in a panel. The objective of this paper is showing an analysis of multilayered discourse using a Japanese Manga work, and discussing layer composition mechanism for introducing the rhetoric into Hyper-comic system.

2. The Outline of Hyper-comic

A basic narrative element in Hyper-comic is "Panel", which means a kind of framework on where a "view(s)" is (are) putted. A view means really described image. Each panel has unique information such as panel's view, panel number, importance, and viewpoint. Panels are mutually connected by temporal order that each element occurs in a story and other narrative discourse techniques based on the systematization of Genette's original typological theory. Genette made a hierarchical classification of narrative discourse rhetoric including three higher categories ("tense", "mood", and "voice") and many lower categories. Tense means temporal relations between a story (an aspect of what is narrated) and really narrated text. As subcategories, tense has "order", "duration", and "frequency". Mood is controlling the proposition of story's information by perspective or focalization and distance between the story world and the narrator. Voice treats problems of narrator and narratee. On narrator, it indicates the spatial level and the temporal level at where a narrator narrates a story. Panels and their links are constructed based on these discourse techniques. In a position of a story, in the case that we can move another position according to a discourse technique, we can prepare a panel and a link making the course. For example, an event which proposes a hint about the past is connected to the past panel with "analepsis" link.

There are two technical mechanisms implementing narrative discourse techniques on Hyper-comic. (1) "Reading course" is related to relations among panels, has two concrete mechanisms; "fixed link" and "automatic editing". The former is a previously fixed network among panels, and the latter automatically rearranges the storing of panels responding to reader's requirement. (2) "Layer composition" is the mechanism which operates panel's view. In hyper-comic, each panel is constructed by the superposition of some layers (figure.1). We can think various types of layers;

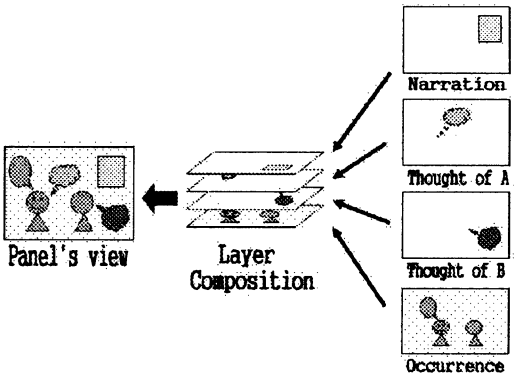


Figure.1 The image of layer composition

external action of a character, thought and mental behavior of a character, narrator's direct utterance or narration, and so on. We can implement narrative

discourse techniques using the layer composition mechanism. For example, internal perspective is constructed by superposing a focalized character's external action and its thought.

Table.1 shows the correspondence between narrative rhetoric and implementation techniques. Left column shows three implementation techniques. Center column shows upper categories of narrative techniques. Right one shows more concrete techniques.

Table.1 Narrative rhetoric and implementation techniques

Fixed inks	Tense	Order (analepsis, prolepsis) Duration (scene, pause) Frequency(singularizing, repeating, iteration)
	Voice	Narrative level (metalepsis)
	The focus of description	The criterion of focus
Automatic Editing	Tense	Order (analepsis, prolepsis) Duration (summary, ellipsis) Frequency (singularizing, repeating, iteration)
	Mood	Perspective (fixed, variable, multiple)
	The focus of description	The operation of description (angle/ distance/ accuracy of description) The criterion of focus (fixed/ variable/ multiple focus of description)
	The rhetoric of panels	Parallel comparison
Layered composition	Tense	Frequency (singularizing, iteration)
	Mood	Perspective (internal/ external/ zero focalization) Distance (narrator's reference)
	Voice	Narrative level (metalepsis) Time of the narrating
	The focus of description	The operation of description (accuracy of description)
	The rhetoric of panels	Distinction of narrative level, Adjustment of time, Adjustment of space
	The rhetoric of characters	Font
	Multilayered discourse	Different temporal level, Different focus of description, Different narrative level

3. An Analysis of Manga and Multilayered Discourse

Based on a rhetorical analysis of Manga, we have found some interesting techniques not including Genette's theory. In this section, we focus on "the focus of description" and "multilayered discourse".

3.1. The method of analysis

We selected "Maeson Ikkoku" [4], which is a famous and long (about 3,500 pages) Japanese comic. First, we picked up about 120 panels including unique and interesting discourse techniques about relations between a panel and another panel. We abstracted about 40 types of techniques. Second, we compared them with Genette's discourse techniques, and especially paid on attention to techniques that Genette's did not explicitly describe. We divide them into next categories; "the focus of description", "the discourse of panel", "the discourse of character", and "multilayered discourse". The multilayered discourse is a kind of synthetic rhetoric which uses simultaneously various discourse techniques. Here, before we explain about it, we refer to the focus of description which has an important relation with the multilayered discourse.

3.2. The focus of description

This is a rhetorical category about the focalization to an object and some kinds of operations of focalization. It is different from Genette's perspective or focalization. In the point that Genette's concept means the viewpoint through which information in a story is proposed. The focus of description has two aspects; "the criterion of focus" is an aspect on what is currently focalized target to be centrally described (including the changing from a focalized target to different target), and "the operation of description" is an aspect on how to move on the focalized target.

The criterion of focus decides which object is a main descriptive target in objects on a panel. In Manga, this target frequently changes or moves to another target when the story goes to a next panel. In the changing in micro level, for example, the criterion of focus moves to a different object in a same event. In macro level, for example, the criterion of focus moves to an object in another event. Figure.2 shows an example that targets of description change to other objects in a same event. In first right panel, three characters are simultaneously described. In next center panel, a woman with black clothes is focused on. In third left panel, another man is described as a focalized criterion. The criterion of focus has three tactical subcategories. In "fixed focus of description", a focalized target does not change while a sequence of an event or a story continues. In "variable focus of description", focalized-targets change in an event or a story. In "multiple focus of description", plural objects equally become focalized targets. In the case of variable focus of description, along temporal progress of an event, focalized targets change, but in the case of multiple focus of description, in a same moment, focalized targets alter.

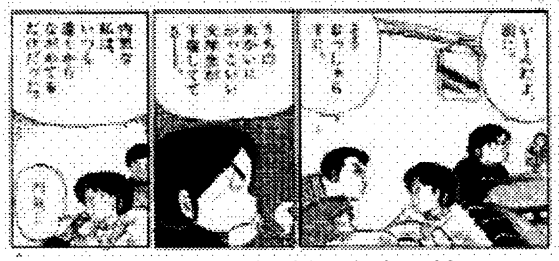


Figure.2 The example of the criterion of focus

The operation of focus, which means techniques to control the focus on a focalized object, is a similar concept with a kind of camera work. It has three subcategories. "Angle of description" shows the angle that a focalized target is seen or described. "Distance of description" decides the distance like near or far that a

focalized target is seen or described. In figure.3, under panel has nearer distance of description than upper panel in a same situation. "Accuracy of description" shows the degree of precision that a focalized target is seen or described. In figure.3, the television's display in under panel has higher degree of accuracy because picture is describing that we can not see in upper television.

Figure.3 An example of distance of description

3.3. Multilayered discourse

We found panels on which different discourse techniques are simultaneously described. For example, figure.4 shows an example that three different criterion of description are appeared in a panel; the external description of an apartment, a sleeping man on the bed (in a room inside the apartment), and the content of his dream. We think that such special and a little bizarre rhetoric is constructed by the superposition of different views in a panel. We call this kind of discourse techniques multilayered discourse. We have very rich examples in Japanese girl's comic, and regard it as one of the most original and important from the viewpoint of narrative rhetoric.

Figure.4 An example of some views in a panel

Similar considerations were proposed by Natsume [6] and Sasamoto [7]. Natsume described a feature that a panel is constructed by the systematic composition or collection of partial elements expressing each object, and this indicates the method of layer composition. On the other hand, Sasamoto called a narration form in which some kinds of event lines polyphonically progresses multiple or multi layered narration. Our idea of multilayered discourse is different from these ideas in the point that we are proposing a special form compressing narrative discourse techniques inside one panel.

In the analysis of Manga, next three types of multilayered discourse frequently appeared; "different temporal levels", "different narrative levels", and "different focuses of description". In first multilayered discourse, some events with different temporal levels, (past, present, and future), are expressed in a panel together. The panel of figure.5 simultaneously expresses a present situation in where a man exists and three events he experienced in the past are described. Second multilayered discourse with different narrative levels mixes temporarily or spatially various levels. For example, in figure.4, a real situation and an imagination in dream appear inside a panel at the same time. Third multilayered discourse is mixing different focuses of description, and an event or a situation is expressed based on multiple criteria of focuses. In the case of figure.6, three focuses (the external description of moving car, a woman in the car, and a man and a girl in the car), are described together.

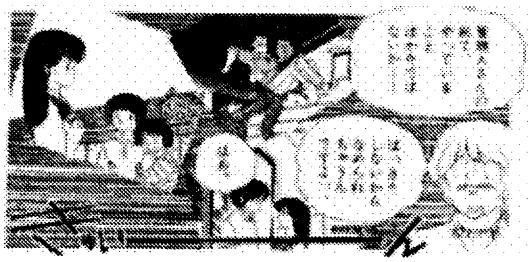


Figure.5 An example of different temporal level

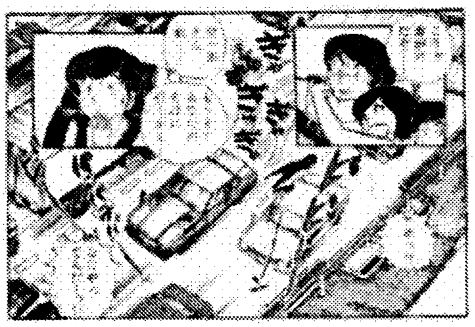


Figure.6 An example of different focus of description

Next, we classify techniques of Manga for concretely constructing multilayered discourse. First type is "the embedding of panels", and a panel is embedded inside another panel. Figure.5 and 6 use this technique. Second type is "the embedding using balloon". In figure.7 a different level's event is embedded into a situation using utterance (or internal monologue) of a character. We tentatively call third type "ground and figure". In the case of this technique, different levels simultaneously appear in background and figure in a panel. In figure.8, a sleeping man in real situation corresponds to figure, and a strange event occurring in his imagination about neighbors corresponds to ground or background. This is a different type of technique from above two ones.

Figure.7 An example of embedding using balloon

Eigure.8 An example of ground and figure

4. An Implementation of Multilayered Discourse

In the basic technical method of Hyper-comic system, each narrative technique, namely techniques acquired by Genette's theory and our Manga analysis, is implemented using the composition of layers as shown in Figure.1. As a complex method of layer composition, we show the basic mechanism.

First, multilayered discourse is used when a reader is reading a panel in Hyper-comic. When the system uses multilayered discourse on a current panel, the system collects a panel or panels which exist(s) in the network in Hyper-comic. This panel or these panels is (are) previously combined with the present panel as a panel or panels allowed the use by multilayered discourse.

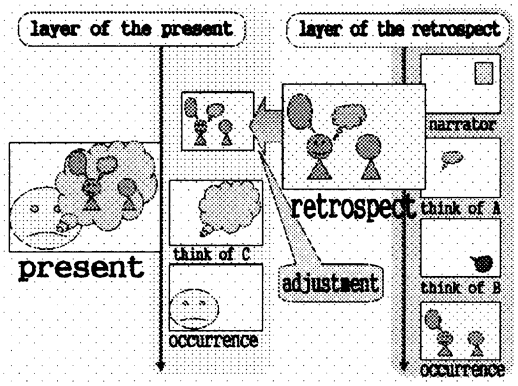


Figure.9 Multilayered discourse by layer composition

Figure.9 shows an abstract layer composition process based on an example of figure.7. In the figure, "retrospective" panel corresponds to a collected panel, and describes an event that a character in the current situation recalls. Of course, this retrospective panel is also previously composed of some layers. Next, the system embeds the retrospective panel into the present panel. In the case of this example, the collected retrospective panel is inserted into the balloon related to a character. In the processing of re-composition of layers, the system adjusts the size of each element. Finally, in the order of layer composition in above example, a retrospective layer was superposed on a present layer with the adjustment of sizes, and a complicated panel was created. The example of figure.6 also takes same order. On the contrary, in the technique of ground and figure like figure.8, a present layer (figure) is superposed on a collected layer (ground) (the strange situation).

5. Conclusions

As a part of the research on Hyper-comic system which is a proposing of a new form of Manga on computer, this paper mainly showed the concept and basic implementation mechanism of "focus of description" and "multilayered discourse" based on analysis of a Japanese Manga work. For operations to control the focus of description, we extracted three types; angle of description, distance of description, and accuracy of description. On the multilayered discourse, we classified its rhetorical techniques into three categories (different temporal levels, different narrative levels, and different focuses of description), and showed some kinds of techniques to construct concretely, (embedding of panels, embedding using balloon, and ground and figure). At last, we simply explained the implementation technique by layer composition on Hyper-comic.

We think that multilayered discourse is the most important and unique narrative discourse technique in Manga. We can implement comparatively easily this technique on Hyper-comic using the method of layer composition which we adopt as one of main implementation technique. Important future works in the research of Hyper-comic is acquiring and classifying various multilayered discourse, and implementing a next version of Hyper-comic with rich fragmentary picture data based on a concrete narrative work and a systematic layer composition mechanism.

References

[1] Endo Y., Ogata T. (2003), Hyper-comic system as consideration of rhetoric, Proceedings of the 4th the International Conference on Cognitive Science, volume1. University of New South Wales Sydney, Australia, July 13-17, 2003, pp.111-116

[2] Endo Y., Ogata T. (2002), Hyper-comic system as representation field of narrative discourse. Proceedings of the 17th Congress of the International Association of Empirical Aesthetics, 4, 8, pp.555-558.

[3] Ogata T. (2002), Expanded literary theory: Cognitive/computational expansion of literary theories and narratology. Proceedings of the 17th Congress of the International Association of Empirical Aesthetics, 4, 8, pp.163-166.

[4] Genette G. (1972), Discourse du recit, essai de methode, In Figures III, Seuil:Paris.

[5] Takahashi R. (1980-1987), Mezon Ikkoku: special wide version collection 1-10 (1992-1993) (in Japanese), Syogakukan, Japan.

[6] Natsume F. (1997), Manga ha naze omoshiroinoka – sono hyougen to bunpou (in Japanese), Nihon-Hoso-Syuppan-kyoukai, Japan.

[7] Sasamoto J. (2002), Shojo-manga ni okeru 「Jyusouteki na katari」 –Tsumugi taku no sakuhin wo rei tosite (in Japanese). Manga Studies, Japan Society for Studies in Cartoon and Comics, Vol.1, pp.121-125.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Implications of the Ability to Learn Simple Actions on the Efficiency of Evolution of Social Behavior of Agents

Ivan Tanev[1], Katsunori Shimohara[1) 2)]

[1)] ATR Human Information Science Laboratories, 2-2-2 Hikaridai, "Keihanna Science City", Kyoto 619-0288, Japan

[2)] Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan

e-mail: {i_tanev, katsu}@atr.co.jp

Abstract

We investigate the effect of ability to learn simple actions on the performance characteristics of evolution of social behavior of agents situated in inherently cooperative environment. Using continuous predators-prey pursuit problem we verified that relatively complex social behavior emerges from simple, implicit, locally defined, and thus – robust and scalable interactions between the predator agents. Considering a distinct aspect of the phenomenon of emergence, we hypothesize that the ability of agents to learn how to perform simple, atomic acts of implicit interaction might facilitate the evolution of more complex behavior. The empirical results indicate that incorporation of the proposed approach of learning in genetic programming (employed as an algorithmic paradigm to evolve the social behavior of the agents) is associated with about two-fold decrease of computational effort of the evolution.

Keywords: emergence, genetic programming, multi-agent system, social behavior

1 Introduction

Over the past few years, multi-agent systems (MAS) have become more and more important in many aspects of computer science such as distributed artificial intelligence, distributed computing systems, robotics, artificial life, etc. MAS introduce the issue of collective intelligence and of the emergence of behavior through interactions between the agents. An agent is a virtual entity that can act, perceive the proximity of its environment and communicate with others; it is autonomous and has abilities to achieve its objectives. MAS contain a world (environment), entities (agents), relations between the entities, a way the world is perceived by the entities, a set of operations that can be performed by the entities and the changes of the world as a result of these actions. Currently, the main application areas of MAS are problem solving, simulation, collective robotics, software engineering, and construction of synthetic worlds [1]. Considering the latter application area and focusing on the autonomy of agents and the interactions that link them together [2], the following important issues can be raised: What is minimum amount of perception information needed to agents in order to perceive the world? What approaches can be applied to automatically construct the agents' functionality, with the quality of such a design being competitive to the design handcrafted by human? Can different approaches be combined to facilitate a superior (in terms of quality or runtime) solution, which is unachievable by each of these approaches? Within the considered context, the *objective* of our research is an efficient automatic design of autonomous agents which situated in inherently cooperative environment are capable of accomplishing complex tasks through interactions. We adhere to the methodological holism based on the belief that any complex system or society [3], and multi-agent society in particular [4] is more than the sum of its individual entities, more than the sum of the parts that compose it. From such a holistic viewpoint we investigate the following relevant aspects:

- The way in which the social behavior, needed to accomplish the complex task emerges in MAS from relatively simply defined interactions between the agents, and
- The effects of the ability of entities in MAS to learn simplest atomic actions on the performance characteristics of the evolution of more complex, social behavior.

Considering the first of these aspects as beyond the scope of current document, we suggest the interested reader to refer to [5] for details. Instead, focusing on the second aspect, we intend to highlight the issues of (i) applying the genetic programming (GP) paradigm for evolving the social behavior of agents and (ii) investigating the effects on ontogenetic learning of simple actions on the performance of such an evolution.

The remaining of the document is organized as follows. Section 2 briefly introduces the task, which we use to test our hypotheses. The same section introduces the software architecture of the agents and briefly elaborates the algorithmic paradigm used to evolve the functionality of agents. The proposed learning mechanism is discussed in Section 3. The same section presents empirical results about effects of learning on the performance of evolution. Section 4 draws a conclusion.

2 Background

2.1 Instance of Predator Prey Pursuit Problem

The general, well defined and well-studied yet difficult to solve predator-prey pursuit problem [6] is used to verify our hypothesis that ability of entities to learn simple actions in MAS might facilitate the evolution of relatively complex social behavior. The evolved social, surrounding behavior emerges from simple, local, implicit, proximity-defined, and therefore – robust and scalable interactions between the predator agents [5]. The problem comprises four predator agents whose goals are to capture a prey by surrounding it on all sides in a world. The world is a simulated two-dimensional continuous torus. Moving abilities of four predator agents are continuous too. We

introduce a proximity perception model for predator agents in that they can see the prey and *only the closest predator agent*, and only when they are within the *limited range* of visibility of their simulated sensors. Prey employs random wandering if there is no predator in sight and a priori handcrafted optimal escaping strategy as soon as predator(s) become "visible". The maximum speed of prey is higher than the maximum speed of predator (i.e. predator-agents feature inferior moving abilities). In order to allow for predators to stalk and collectively approach the prey the range of visibility of predators is more than the range of visibility of the prey. We consider this case in order to create an inherently cooperative environment in that the mission of predators is nearly impossible unless they collaborate with each other.

2.2 Architecture of The Agents

We adopted the subsumption architecture of the agents [7] comprising of functional modules distributed in three levels, corresponding to the three different aspects ("levels of competence") of agent's behavior: wandering/exploring, greedy chase and social behavior – surrounding. We focus our attention on the issues of evolving the top-level, highest priority module – surrounding the prey assuming that wandering/exploring and greedy chase are a priori handcrafted and incorporated into the agent's controller. Such an approach allows to simultaneously evolve (i) the capability of agents to resolve social dilemmas, determined by the way social behavior overrides greedy chase when prey is in sight, and (ii) the capability to resolve the exploration-exploitation dilemma, determined by the ability of social behavior to override wandering/exploring when prey is invisible [5].

2.3 Algorithmic Paradigm: Strongly-Typed Genetic Programming

Set of stimulus-response rules is a natural way to model the reactive behavior of predator agents [Holland], which in general can be evolved using artificial neural networks, genetic algorithms, and genetic programming (GP). GP is a domain-independent problem solving approach in which a population of computer programs (individuals) is evolved to solve problems [8]. The simulated evolution in GP is based on the Darwinian principle of reproduction and survival of the fittest. The strength of GP to automatically evolve a set of stimulus-response rules featuring arbitrary complexity without the need to a priori specify the extent of such complexity might imply an enormous computational effort caused by the need to discover a huge search space while looking for the potential solution to the problem. Agreeing with [3] that for huge and multidimensional search spaces the introduction of "pruning algorithms" is a critical step towards efficient solution search, we impose a restriction on the syntax of evolved genetic programs based on some a priori known semantics. The approach is known as strongly typed genetic programming (STGP) [9]. The function set of STGP comprises IF-THEN statement, arithmetical operations and comparison operators. The terminal set features local, proximity defined sensory- and continuous moving abilities [5]. The genetic operations are binary tournament selection, random sub-tree and transposition mutations. The breeding strategy is homogeneous: the performance of single genetic program, cloned to all the agents is evaluated. The fitness of the genetic program is evaluated as average of the fitness measured over 10 different, randomly created initial situations. The fitness measured during the trial starting with particular initial situation accounts for (i) the average energy loss of the agents during the trial, (ii) the average distance of the agents to the prey by the end of the trial, and (iii) the elapsed time of the trial. Smaller values of fitness function correspond to better performing predator agents.

3 Learning to Interact Embedded in Evolution of Social Behavior

3.1 Ability to Learn to Interact

Explicitly referring to the genetic representation of agents accessed during genetic manipulations and evaluation as genotype and phenotype respectively, we view the process of learning as a modification to the phenotype, which yields improved performance of the agents. Motivated by the intention to simulate plausible examples from the nature, we consider a pure Darwinian phenotype plasticity (a case of Baldwinian evolution), which implies that the ontogenetically improved phenotype is not reversed back into the corresponding genotype. We focus our attention on the indirect influence of ability of agents to ontogenetically learn simple actions on the performance characteristics of evolution of more complex, social behavior in MAS. We consider the following aspects of ontogenetic learning via phenotype plasticity: (i) the objective of learning (i.e. learning task), (ii) the way of selecting the phenotypic fragment to be modified, and (iii) the modification algorithm.

Following the intuitive concept that ability of agents to interact with each other might contribute to more quickly attain a better performing social (surrounding) behavior, we define the objective of learning as simple increase of amount of atomic actions related to interactions between the agents. We consider any act of referring to the relative position of the closest peer agent in the world as an implicit interaction between the agents. The amount of references to the sensory variables related to the peer agents are counted during the trial and the difference between the values of the counter after- and prior to the modification of the phenotype is considered as an ability of agents to learn. The overall fitness is calculated as a combination of the fitness, obtained from the (i) behavior of the agents prior to modification of their phenotype and the (ii) ability to learn to interact. The learning algorithm, incorporated into the fitness evaluation routine is shown in Figure 1.

Three ways of selecting the modified fragments in the phenotype have been implemented: (i) random tree node (RM), (ii) tree node from the set of most recently introduced (via mutation) nodes in genotypic representation of the agent (MRM), and (iii) tree node from the set of least active nodes in genotypic representation of the agent (LAM). The case (ii) is based on the empirical observation that new structures in genotype, introduced by random mutations most likely yield a inferior phenotypic structures, while (iii) relies on the hypothesis that the amount of interactions can be

increased by altering the less active, or ultimately, the totally inactive fragments in phenotypic representations of the agents. The algorithm of modification of selected phenotypic fragment is sub-tree random mutation. Both the selection and modification mechanism are implemented in routine `Modify(OriginalGP)` shown in Figure 1, Line 14.

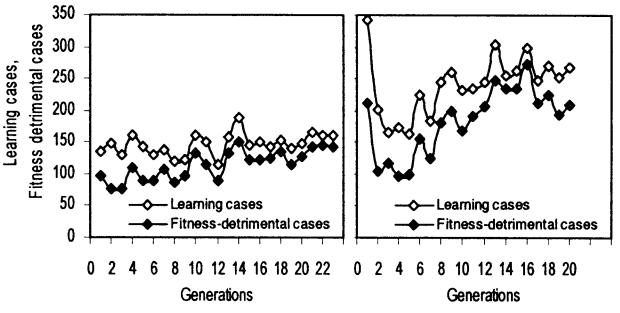
```
 1. Procedure EvalWithLearning(OriginalGP: TypeGP;
 2.                    LearningCycles,
 3.              [out] Fitness,
 4.              [out] SuccessfulSituations:integer);
 5. var
 6.   GP: TypeGP;
 7.   i, OrigAmountOfInter, AmountOfInter: integer;
 8. begin
 9.   Clone_GP_To_All_Agents(OriginalGP);
10.   Evaluate_Behavior([out] Fitness,
11.                     [out] SuccessfulSituations,
12.                     [out] OrigAmountOfInter);
13.   for i:=0 to LearningCycles-1 do begin
14.     GP := Modify(OriginalGP);
15.     Clone_GP_To_All_Agents(GP);
16.     Evaluate_Behavior([out] AmountOfInter);
17.     if AmountOfInter > OrigAmountOfInter
18.       then Fitness :=   Fitness
19.            - dF(AmountOfInter-OrigAmountOfInter);
20.   end;
21. end;
```

Figure 1. The learning algorithm

3.2 Empirical Results

The parameters of STGP used in our experiments are as follows: the population size is 400 genetic programs, the selection ratio is 0.1, and the mutation ratio is 0.02. The termination criterion is defined as a disjunction of the following conditions: (i) fitness of the best genetic program in less than 300 and the amount of initial situations in which the prey is captured equals 10 (out of 10), (ii) amount of elapsed generations is more than a 100, and (iii) amount of recent generations without fitness improvement is more than 16. The raw fitness value of 300 corresponds to the successful team of predator agents, which, averaged over all initial situations, captures the prey approximately by the middle of the trial interval (limited to 600 time steps). Single learning iteration is performed (value of parameter `LearningCycles` depicted in Figure 1, Line 13 equals 1). Superior sensory abilities of predators and inferior moving abilities are considered.

The dynamics of (i) the amount of learning cases per generation (the modified phenotype of individual features increased interaction activity) and (ii) the amount of fitness detrimental cases, i.e. cases when improved interaction abilities of phenotype would result in inferior fitness if reversed back into the corresponding genotype for learning via RM and LAM are shown in Figure 2a) and Figure 2b) respectively. The analogical characteristics of learning via MRM are essentially the same as in the case of RM. As figure illustrates, learning with LAM is characterized by increased overall amount of learning cases. Despite the fact that LAM features similar to RM ratio of fitness detrimental cases to overall learning cases (which is relatively high), the absolute difference between the overall amount of learning cases and fitness detrimental cases remains relatively high, indicating a possible dynamic correlation between the learning and the fitness landscapes. Computational effort (the amount of individuals needed to be evaluated in order to obtain the solution with specified probability, e.g. 0.95) of four cases is obtained from the probability of success `p(t)` by each of 20 independent runs in a way as suggested in [8]. The four cases correspond to the (i) evolution without learning, evolution influenced by learning via (ii) RM, via (ii) MRM, and via (iii) LAM respectively. The results, shown in Figure 3 indicate p(t)=0.95 for of about 32,000 evaluated individuals for evolution without learning verifying the very concept that relatively complex social behavior is evolvable and it emerges from simple, implicit, locally defined interactions between the predator agents.



a) b)

Figure 2. The amount of learning cases and fitness detrimental cases per generation for learning via RM (a) and LAM (b) respectively

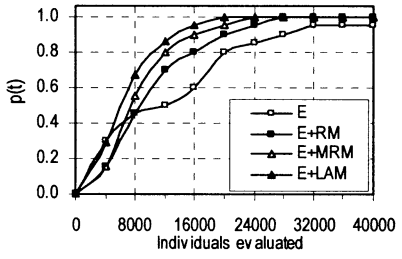


Figure 3. Probability of Success in Evolving Social Behavior of the Agents

The best computational effort of about 16,000 individuals (including the computational cost of learning – the amount of individuals, additionally evaluated during the learning cycle as shown in Figure 1, Line 16), achieved in case of evolution influenced by learning via LAM (denoted as E+LAM), indicating about two-fold improvement of computational effort of STGP. The assumed dynamic correlation of the learning landscape [10] with the probability of being closer to the optimal solution is considered as a primary reason for observed improved computational effort of evolution.

Trace of the entities in the world, where the team of predator agents is governed by sample best-of-run genetic program in one of considered initial situations is shown in Figure 4. The prey, originally situated in the center of the world, is captured by time step 118. The emergence of following behavioral traits of predator agents are noticeable: (i) switch from greedy chase into surrounding approach of agents #2 (time step 65) as soon as other agents appear in sight; (ii) zigzag move by agent #1 which results a lower chasing speed indicating "intention" to trap the prey (after time step 40) and (iii) surrounding approach demonstrated by agents #0, #3 (top) and #2 during the final stages of the trial.

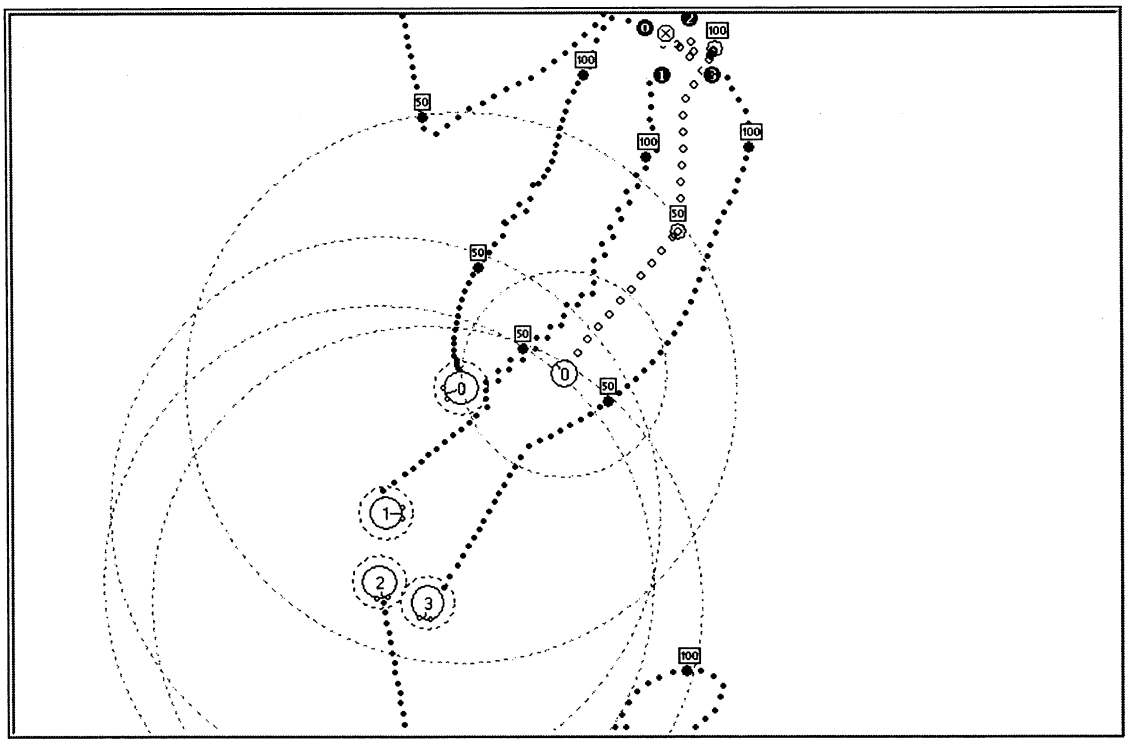


Figure 4. Traces of the entities with predator agents governed by sample best-of-run genetic program. Prey is captured in 118 simulated time steps. Larger white and small black circles denote the predator agents in their initial and final position respectively. The small white circle indicates the prey, initially situated in the center of the world. The numbers in rectangles show timestamp information

4 Conclusion

We presented the result of our work on the effect of learning to interact on the evolution of social behavior of agents situated in inherently cooperative environment. Using continuous predators-prey pursuit problem we verified that relatively complex social behavior emerges from simple, implicit, locally defined, and therefore – robust and scalable interactions between the predator agents. In addition we hypothesized that the ability of agents to learn to perform simple, atomic acts of implicit interaction with each other facilitate the performance of evolution of more complex, social behavior. The empirically obtained results indicate that incorporation of the proposed approach of learning in genetic programming is associated with about two-fold decrease of computational effort of the evolution.

Acknowledgements

This research was conducted as part of "Research on Human Communication" with funding from the Telecommunications Advancement Organization of Japan.

References

[1] J.Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, Harlow: Addison Wesley Longman, 1999

[2] H.Parunak, S.Brueckner, M.Fleischer, J.Odell, Co-X: Defining what Agents Do Together, Proceedings of the AAMAS 2002 Workshop on Teamwork and Coalition Formation, Onn Shehory, Thomas R. Ioerger, Julita Vassileva, John Yen, eds., Bologna, Italy, 2002

[3] H.J.Morowitz, The emergence of Everything: How the World Became Complex, New York: Oxford University Press, 2002

[4] J.H.Holand, Emergence: From Chaos To Order, Cambridge: Perseus Books, 1999

[5] I.Tanev, and K.Shimohara, On Role of Implicit Interaction and Explicit Communications in Emergence of Social Behavior in Continuous Predators-prey Pursuit Problem, Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2003) Vol 1, 74-95, 2003

[6] M.Benda, B.Jagannathan, and R.Dodhiawala, On Optimal Cooperation of Knowledge Sources. Technical Report BCS-G2010-28 Boeing AI Center, Boeing Computer Services, Bellevue,WA, 1986

[7] R.Brooks. A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation 2(1): 14-23, 1986

[8] J.R.Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, Cambridge, MA: MIT Press, 1992

[9] D.Montana, Strongly Typed Genetic Programming, Evolutionary Computation 3(2):199-230, 1995

[10] D.Floreano, S.Nolfi, F.Mondada, Co-evolution and ontogenetic change in competing robots,In J.P.Mukesh, V.Honavar & K. Balakrishan (Eds.), Advances in Evolutionary Synthesis of Neural Networks, 273-306. Cambridge, MA: MIT Press, 2001

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

An Immunity-Based Approach towards a Synthesis of Artificial Organisms

T. Mori Y. Ishida
Department of Knowledge-Based Information Engineering
Toyohashi University of Technology
Toyohashi, Aichi, Japan 441-8580

Abstract

We propose a model and multi-agent simulations towards a synthesis of an artificial organism. The model consists of autonomous agents with types. In the simulation, two 2-dimensional lattice spaces celled *Agent Space* and *Shape Space* are introduced. Each agent is placed at a site in *Agent Space* and can move randomly in the space. The type of each agent is specified by a site in *Shape Space*, and the type of each agent is fixed for the agent lifetime. Agents interact with each other: suppression or stimulation depending on positions in *Agent Space* and types in *Shape Space*. The following three life-like phenomena are observed by the multi-agent simulation. First, clusters of agents are formed in both *Agent Space* and *Shape Space*. This cluster formation could be understood as a primitive form of *innate Immunity*. Second, two clusters can be fused or one cluster destroys another cluster depending on the course of development. Third, host-parasite relation between clusters of agents emerges.

1 Introduction

We present a model and multi-agent simulations towards a synthesis of an artificial organism. Throughout the paper, we used an immunity-based approach, since the immune system is intrinsic to the multicellular organism not only for providing a protective function but for a boundary formation of "the self".

Boundary formation is an important emergent phenomenon in *Autopoietic Systems*[4]. With a self-organizing view on the immune system, "self-assertion" has been proposed as an emergent property intrinsic to the immune system. Bersini proposed a CA-like model[3] where each site has a specificity in interacting with other sites in a space similar to "shape space". In his simple model, not only "self-signature" emerged but immunologic phenomena such as immunologic memory, immunologic tolerance and non-self elimination are also explained.

In our model, *Agent Space* as well as *Shape Space* is introduced where physical position of each agent is specified in *Agent Space* and the type of each agent in *Shape Space*.

On the other hand, synthetic approaches to artificial organism have been done in Artificial Life (e.g., [1], [2]). Emergence of self-reproduction and ecological interaction may be conditions for organisms, where the former is studied by Langton[1] and the latter by Ray[2].

Our model does not consider the self-reproduction of organisms, however, several ecological interactions such as the host-parasite relation have emerged in simulations. Further, agents form clusters in *Agent Space* where the type of agents in a cluster in *Shape Space* remains invariant.

2 The Basic Model

The model consists of autonomous agents where each agent mounted a receptor for an interaction with specific agents. This specificity is determined by the type identified in *Shape Space*. Each agent is assumed to be capable of:

- Replicating agents whose type (receptor) is the same as (or similar to in case of mutation) that of the generator;
- Suppressing (or stimulating) other agents with a specific type hence causing them to decrease (or increase) in numbers;
- Moving in *Agent Space* allowing them to form a cluster and to fuse with other clusters.

We use two 2-dimensional lattice spaces:

- *Shape Space* expressing the "affinity" between two agents. That is two agents placed close in this space have a similar type and hence a high affinity in their receptors.

- *Agent Space* expressing a physical layout of agents. That is, two agents placed in the neighbor can interact with each other. A physical sense of space is realized by *Agent Space* where agents can move randomly.

Both *Shape Space* and *Agent Space* is realized by a 2-dimensional lattice space with a periodic boundary condition. The size of *Shape Space* is 60x60, and that of *Agent Space* 200x200 in simulations.

Further, each agent has the following parameters:

- *Concentration* (c): Agent concentration indicates the number of agents with the same receptor. In the simulation, concentration is indicated by the size of agent in *Agent Space*. The number of initial concentration $c(0)$ for each agent is set to be 100 in all the simulations.
- *Birth Rate* (b), *Death Rate* (d), *Mutation Rate* (m): Agent concentration c will increase with a rate b and decrease with a rate d. When mutation occurs with a rate m, concentration of agent in the *Moore* neighbor in *Shape Space* will increase. These parameters are common to all the simulations.
- *interaction Threshold* (T), *Contact Threshold* (K), *Interaction Intensity* (P): Any pair of agents who are within a distance K in Agent Space can interact. Agents will suppress (or stimulate) other agents when the distance between them exceeds (or does not exceed) T in *Shape Space*. The fraction that agent suppress (or stimulate) the other agent is in proportion to the distance between the agents. Suppression (or Stimulation) means decreasing (increasing) the concentration (c) of the other agent with an intensity (p).

At every time step, the agent i causes to increase (or decrease) concentration of the agent j in the following way:

$$c_j(t+1) = c_j(t) + P \times (T - d)/L \quad (d \equiv dist(i,j))$$

When the distance $dist(i,j)$ between agent i and j is smaller that the *Interaction Threshold* (T), the concentration of agent j increase, otherwise it decrease. L is set to be 100 when $d \geq T$ to reflect the low affinity, and set to be 1 otherwise. The agent with a negative concentration is eliminated from the space.

3 Simulation Results

The following life-like phenomena are observed by starting with different initial configuration of agents: cluster formation, fusion (and rejection) of clusters and an emergence of host-parasite relation between clusters.

3.1 Cluster Formation

Simulations started from a random initial configuration both in *Shape Space* and *Agent Space*. Clusters are gradually formed in *Shape Space*. After some time steps, clusters often become stable (Fig.1:Left). When an agent is added to a place (in *Shape Space*) whose distance exceeds T, the agent will promptly be removed. Thus, the cluster may be regarded as "the self" which has a stable set of types in *Shape Space*.

This phenomenon of cluster formation could be understood as a primitive form of *Innate Immunity* found in all the animals.

Following after the cluster formation in *Shape Space*, clusters are formed in *Agent Space* as well.

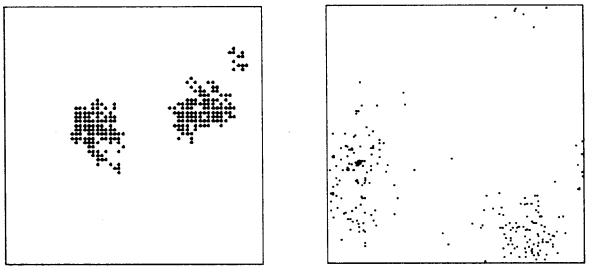


Figure 1: Left:*Shape Space*; Right:*Agent Space*, 5000 time steps after 400 agents are placed randomly in *Agent Space* and their types are also randomly distributed in *Shape Space*. Parameters are: $T = 8, P = 1.0, b = 0.1, m = 0.03, K = 8$ and $d = 5$.

3.2 Fusion and Rejection of Clusters

Simulations started from an initial configuration with two clusters placed in *Shape Space* so that the distance between centers of the clusters is $2T$ (Fig.2). In *Agent Space*, all the agents are placed within the region (e.g., a circle with the radius K) that each agent can mutually interact. Then, it is observed that two clusters can be fused (Fig.3) or one cluster destroys another cluster (Fig.4) depending on the course of development.

Next, we conducted simulations with a random configuration of initial agents both in *Shape Space* and *Agent Space*. Agents first form several clusters, then after repeatedly fusing or destroying other clusters, and finally end up with few clusters (Fig.5).

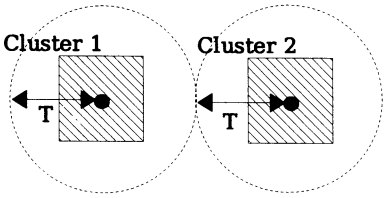


Figure 2: Distance between two clusters is $2T$ in *Shape Space*. ($T = 5$)

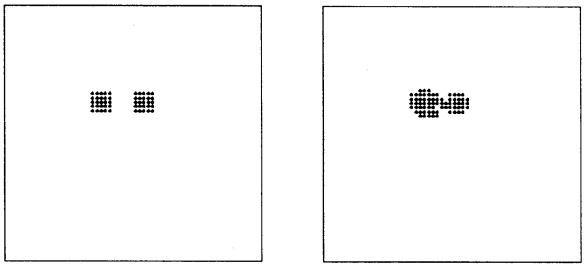


Figure 3: *Shape Space* at time step 0 (left) and 2000 (right), Coordinates of the center of initial 5x5 clusters are:(22, 22), (32, 22). Parameters are: $T = 5, P = 1.0, b = 0.1, m = 0.03, K = 50.0$ and $d = 5$.

3.3 Host-Parasite Relation Between Clusters

Host agents ($T_H = 20$) and parasite agents ($T_P = 0$) are considered. Since the *Interaction Threshold* for parasite agents T_p is 0, this agent will suppress any other agents. Simulations started from an initial configuration where two clusters placed in *Shape Space* as shown in Fig.6. Then, simulations show that host-parasite relation would emerge. Parasite agents cannot live without host agents. That is, the concentration of the host agents are partially decreased by the parasite agents, while that of the parasite agents is increased by the host agents.

Fig.7 shows a placement of a host cluster (cluster H) and a parasite cluster (cluster P) in the *Shape Space*.

Since agents in the cluster P are within the Interaction Threshold of the cluster H, their concentration will be increased.

Parasite agents do not increase explosively in simulations. Fig.8 shows time evolution of the proportion of population of the parasite agents. The proportion converges on a constant value (0.2) which depends

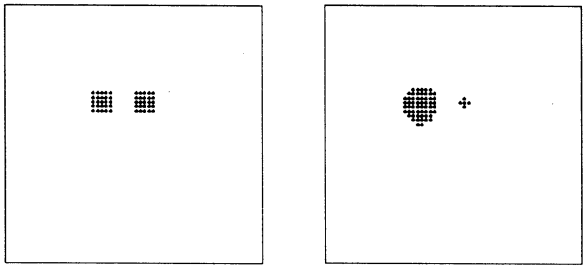


Figure 4: *Shape Space* at time step 0 (left) and 1800 (right), Coordinates of the center of initial 5x5 clusters are:(22, 22), (32, 22). Parameters are: $T = 5, P = 1.0, b = 0.1, m = 0.03, K = 50.0$ and $d = 5$.

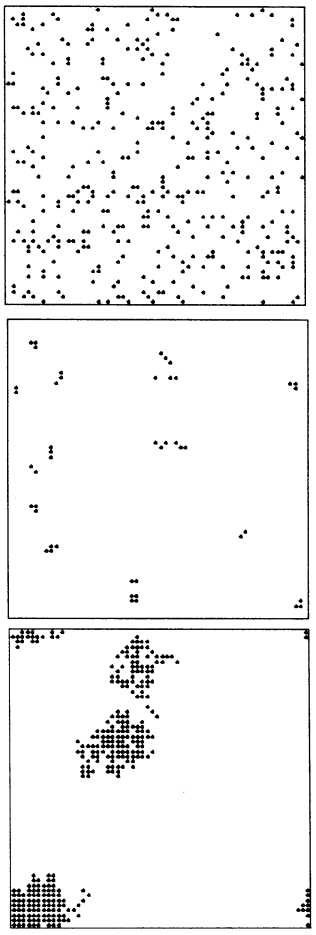


Figure 5: *Shape Space* at time step 0 (above), 100 (middle) and 6000 (below) after 400 agents are placed randomly in *Agent Space* and their types are also randomly distributed in *Shape Space*. Parameters are: $T = 5, P = 1.0, b = 0.1, m = 0.03, K = 10.0$ and $d = 5$.

on the parameters: *Interaction Threshold T*, *Contact Threshold K* and *Interaction Intensity P*.

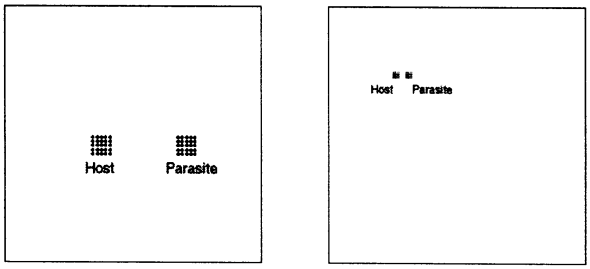


Figure 6: Left:*Shape Space*, Right:*Agent Space*. Coordinates of the center of initial 5x5 clusters are:(22, 32), (42, 32) in *Shape Space*, (52, 52),(62, 52) in *Agent Space*. Parameters are: $T_H = 30, T_P = 0, P = 1.0, b = 0.1, m = 0.03, K = 10.0$ and $d = 5$.

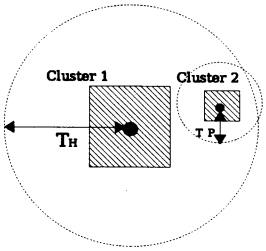


Figure 7: Placement of a host cluster and a parasite cluster in *Shape Space*.($T_H = 20, T_P = 0$)

4 Conclusion

With multi-agent simulations, we observed that agents form a cluster that may roughly correspond to an organism in the following points:

- The cluster maintain a boundary both in *Agent Space* and in *Shape Space* by rejecting internal and external challenges of agents with different type.
- The cluster can have an ecological relation (such as host-parasite) with other clusters.

However, these clusters lack several properties such as self-reproduction, ontogenesis and functional specialization, which seem to be essential in biological organisms. These properties may require a modification of the model, and left for the future study.

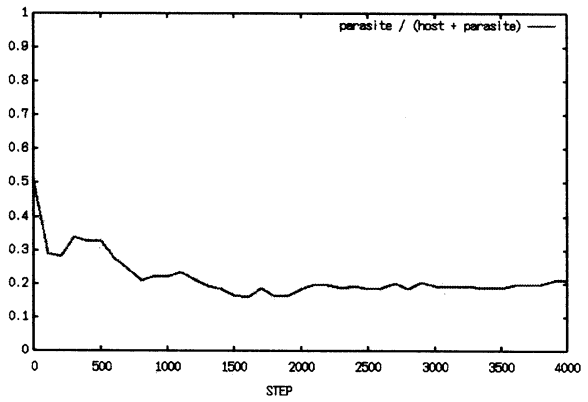


Figure 8: Time Evolution of the Proportion of Population of the parasite agents

Acknowledgements

This work was partly supported by the 21st Century COE Program "Intelligent Human Sensing" from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] C.G.Langton, "Studying Artificial Life With Cellular Automata." *Physica* 22D, pp. 120-149, 1986.

[2] T.Ray, "An Approach to the Synthesis of Life," *Artificial Life II*, SFI Studies in the Sciences of Complexity, Vol. X, pp. 195-226, 1991.

[3] H.Bersini, "Self-Assertion versus Self-Recognition: A Tribute to Francisco Varela," in J.Timmis, and P.J.Bentley, (eds). *Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS-2002)*, University of Kent at Canterbury Printing Unit, pp. 107-112, 2002.

[4] B.McMullin, and F.G.Varela, "Rediscovering Computational Autopoiesis," 1997.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Fault-Tolerance in Biological Systems Simulated on Asynchronous Cellular Automata

Ferdinand Peper*,†, Teijiro Isokawa†, Fukutaro Abo†, Jia Lee*,
Susumu Adachi*, Nobuyuki Matsui†, Shinro Mashiko*

Abstract

Biological systems are remarkably robust against faults. They are able not only of conducting self-repair of structural defects, but also of on-the-spot corrections of transient faults. This paper studies the latter type of fault correction. To this end, we describe a cellular automaton (CA) in which cell states are subdivided in partitions, each of which can be corrected locally by a comparison with a set of four correct patterns. We use an asynchronous timing model for the CA, because of its biological plausibility.

1 Introduction

The ability to self-repair structural defects and to correct transient errors are an important feature of biological systems. Understanding it is instrumental in the realization of next-generation computers with feature sizes on nanometer scales. The increasing interest in fault-tolerance of computers modeled after nature has led to proposals of hardware that can reconfigure itself such as to circumvent defects. For example, in the CA implementation in [1] cells are able to skip defect neighbors, and in the Teramac computer [2] defect components are avoided by adapting the table of routes between components. Though this strategy of making roundabouts around defects is successful in creating reliable computer systems, biological organisms can do even better: they can actively repair defects in their DNA, using redundant information encoded in the DNA. Transient errors (caused by for example noise) are also coped with with a remarkable efficiency, witness for example neural processing in the brain, which is subject to high degrees of noise, but which still operates virtually error-free.

*Communications Research Laboratory, Nanotechnology Group, 588-2 Iwaoka, Iwaoka-cho, Nishi-ku, Kobe, 651-2492, Japan, Email:{peper,lijia,sadachi,mashiko}@crl.go.jp

†Himeji Institute of Technology, Division of Computer Engineering, 2167 Shosha, Himeji, Hyogo, 671-2201, Japan, Email:{isokawa,fabo,matsui}@comp.eng.himeji-tech.ac.jp

This paper aims to shed more light on the mechanisms that play a role in the correction of transient errors. It employs a CA in which each cell is divided into four partitions, each containing the equivalence of a few bits of information. Each partition is encoded by a simple error correcting code containing a small number of code words (four in our case). This results in CAs of which up to one third of the information in a cell may be corrupted without affecting the ability of correcting it. We propose a correction mechanism in which each partition is compared to a small set (four in our case) of correct patterns, and which allows for a straightforward correction. The mechanism also allows for the correction of up to half the errors in a cell in a large number of cases that we shall identify in more detail. The model employs an asynchronous mode of timing, that is, all cells undergo transitions at times that are random and uncorrelated to each other. Compared to the commonly used synchronous timing model, this mode of timing resembles more closely the interactions on microscopic scales, and may thus be considered biologically more plausible (see also [3]).

In the context of CAs, early work on fault-tolerance is reported in [4]. This model can correct at most one error in 19 cells, but to this end each cell needs read access to the states of 49 cells in its neighborhood, which is much higher than the four cells usual in CAs (von Neumann neighborhood). The increased complexity of cells makes them very error-prone in physical implementations: consequently, this work is mainly of theoretical value. The model in [5] suffers from similar problems. Better fault-tolerance is obtained in [6, 7, 8] with synchronous CAs, and in [9, 10] with asynchronous CAs simulating synchronous CAs: the idea is to organize cells in blocks that perform a fault-tolerant simulation of a second CA, which on its turn is also organized in blocks, simulating even more reliably a third CA, and so on. This results in a hierarchical structure with high reliability at the higher levels. The cells in these models are very complicated, however, since they contain information regarding the hierar-

chical organization, such as block structure, address within a block, programs selecting transition rules, and timing information. In [11] fault-tolerance is implemented in a CA by interpreting the state of each cell together with the states of its neighboring cells as a code word in an error correcting code. This requires the states of cells and their neighbors to be updated synchronously. The drawback of this method is that transition rules need to have a certain "minimum distance" to each other with regard to their left hand sides to make error correction possible. This limits the range of possible transition rules and cell configurations. A similar drawback applies to the fault-tolerant asynchronous CA in [12], which is based on *BCH codes* (e.g. [13]). Moreover, computation on this model is inefficient, since only one signal at a time is allowed. An improvement on this model is proposed in [14], and this is explored in more detail in the current paper.

2 Asynchronous Cellular Automata

A CA is an array of identical cells, each of which has four neighboring cells, at its north, east, south, and west. Only two-dimensional CAs are considered here. We will use a CA that is particularly useful for asynchronous updating. Each side of a cell in this model has a memory of a few bits attached to it (Fig. 1). The four memories at the side of a cell are said to

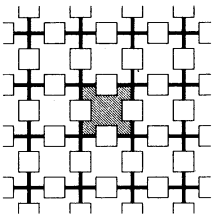


Figure 1: Cellular automaton consisting of cells, one of which is shaded. Each cell has access to four memories, depicted by the small squares. A memory is shared by two cells and it stores a few bits of information.

be *associated* with the cell. The *state* of a memory is the value stored in it. The assignment of a particular combination of states of the memories associated with a certain set of cells is called a *configuration*. A cell may change the states of the four memories associated with it according to an operation called a *transition*. The transitions a cell may undergo are defined by the CA's *transition rules*. A transition rule applies to a cell if its left hand side matches the combination of states of the memories associated with the cell. The

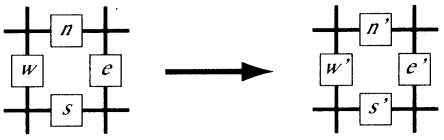


Figure 2: Transition rule describing a transition of the memories associated with a cell. If the memory states of a cell match the states n, e, s, and w in the left hand side, the rule may be applied to the cell, as a result of which the states are changed into the states n', e', s', and w', respectively, in the right hand side.

rule's right hand side describes into what states the memories are changed when it is applied (Fig. 2). As a memory is shared between two cells, it may be updated by a transition acting on either of these cells. Certain desired behavior of a CA, like universal computation [14, 15], can be obtained by setting its memories in proper states and defining transition rules that lead to state changes corresponding to this behavior.

The CA in this paper is *asynchronous*, i.e., a transition of a cell occurs at random times, independent of the transitions of other cells (see also [16, 17, 18, 19]). A transition of a cell only takes place when the cell's memory states match the left hand side of a transition rule. If there is no transition rule to which a match can be made, the cell will not undergo a transition. Two neighboring cells are not allowed to undergo transitions simultaneously. This ensures that the cells will not attempt to set the memory shared between them to different states at the same time, thus preventing write conflicts that could lead to undefined memory states.

The non-fault-tolerant version of the asynchronous CA in this paper has memories of two bits each, a model also known under the name *Self-Timed Cellular Automaton (STCA)* [14, 15, 17, 20]. Interconnection lines in this model take the form of straight continuous areas of cells, called *paths*. When there are no signals on a path, all corresponding memories are in state 0. A signal on a path is represented as a cell in which three of the cell's memories are in state 0 and one memory contains one 1-bit and one 0-bit. Fig. 3(a) shows an example of a transition rule that, operating on a signal, moves the signal's 1-bit southwards by one cell. Applying the transition rule twice gives rise to the sequence of configurations in Fig. 3(b) with the 1-bit propagating southwards along a path of cells. The rotated and reflected equivalents of transition rules may also serve as transition rules. This allows the above transition rule to be used for transmitting signals in

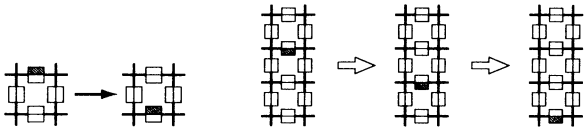


Figure 3: (a) Transition rule for signal propagation, and (b) its two-fold application to a configuration of cells, each time moving a signal southwards by one cell. A memory contains two bits, each of which is indicated by a block that is shaded for the value 1 and white for the value 0.

$$\begin{array}{lll} & \overbrace{}^{a} & \overbrace{}^{b}\ \overbrace{}^{c} \\ \text{Code word 1} = & 0...0 & 0...0\ 0...0 \\ \text{Code word 2} = & 1...1 & 1...1\ 0...0 \\ \text{Code word 3} = & 0...0 & 1...1\ 1...1 \\ \text{Code word 4} = & 1...1 & 0...0\ 1...1 \end{array}$$

Figure 4: Scheme of a code with four code words, one of which contains all 0 bits. Each bit position (column) contains two 0-bits and two 1-bits. The bits are arranged in three different groups a, b, and c, whereby all bits of a word within a group are identical.

directions towards the north, east, or west as well.

We have thus defined a model in which each cell performs transitions on information stored in the memories associated with it. To compute on this model, additional transition rules are necessary, like the ones in [14, 15]. In the next section, we equip the memories with error correcting ability by adding bits to them that store redundant information.

3 Fault-Tolerance

The construction of error correcting codes is based on adding redundancy to information to make it more robust to errors occurring in, for example, transmissions due to noise. The standard way to accomplish this is by defining a set of code words each encoding a block of message symbols, such that if up to a certain number of bits of a code word is corrupted, they can be restored to their original. A set of codewords is called a *code*. The degree to which a code is tolerant to errors is determined by the properties of the code. Three important properties of a code are its *word length* n, expressed as bits, the *number of code words* m, and the *minimal distance* d between any two code words, i.e., the minimum number of bits by which two code words differ. A code is usually denoted as a 3-tuple (n, m, d). If a code has a minimum distance d, it can *correct* up to $\lfloor (d-1)/2 \rfloor$ errors. To design an efficient error correcting code given a certain value of n, the value of d and m should be as large as possible, but these are conflicting requirements: larger distances tend to give rise to fewer code words and vice versa. We will assume codes to be *linear*. Linear codes have the property that the bit-wise addition—modulo 2 in case of binary codes—of any two codewords produces again a codeword. The all-zero code word is always contained in a linear code, because it can be obtained by subtracting any code word from itself.

Our CA is made fault-tolerant by encoding each memory of a cell by an error correcting code [14]. We start with codes that can correct up to one bit error. The messages to be encoded by an error correcting code are all the possible states of the 2-bit memory, i.e., the four memory states 00, 01, 10, and 11. This translates into four code words, so $m = 4$. A code able to correct one error needs to have a minimal distance of $d = 3$. It is shown in [14] that the so-called *Plotkin bound* (e.g. see [13]) imposes a lower bound of $n = 5$ on the word length in this case. Indeed, a (5,4,3)-code exists, for example the code with the words 00000, 11100, 00111, and 11011 (see also [14]). Flipping any arbitrary single bit in any of these code words results in a word that has distance 1 to the original code word, and distance at least 2 to each of the other code words. So, even if one bit is corrupted, it can be uniquely determined what the original code word should have been, enabling us to correct the bit.

To generalize to codes with different lengths that correct more than one error, we observe that codes with a high minimum distance tend to have their 0- and 1-bits evenly distributed among the code words for each bit position. For four code words this translates into each bit being 0 for two code words and 1 for the other two code words. Assuming that one of the code words is the all-zero code word, we then obtain the scheme in Fig. 4. Let the number of bits in groups a, b, and c in Fig. 4 be n_a, n_b, and n_c, respectively, then $n = n_a + n_b + n_c$. In [14] it is shown that the minimal distance d of the code is maximized when the differences between n_a, n_b, and n_c are as small as possible. We distinguish three cases, depending on the remainder when n is divided by 3 (see [14]):

- $n = 3p$ for some p. The distance will be maximal if $n_a = n_b = n_c = p$. In this case $d = 2p$, so up to $p - 1$ errors can be corrected in a word.

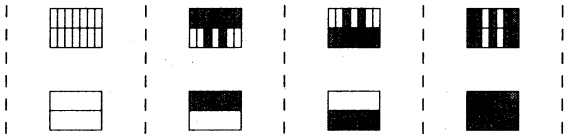


Figure 5: Graphical notation of a (14,4,9)-code. The top row contains the code words, the bottom row the original messages. A dark block encodes a bit with value 1, a white block a bit with value 0. The bits of the code words have been shuffled to make for an easy-to-discern graphical representation. This code allows the correction of up to four arbitrary bit errors.

- $n = 3p+1$ for some p. The distance will be maximal if two groups have length p and one group has length $p+1$, so let $n_a = n_c = p$ and $b = p+1$. In this case $d = n - (p+1) = 2p$, which implies that up to $p-1$ errors can be corrected in a word.
- $n = 3p+2$ for some p. The distance will be maximal if two groups have length $p+1$ and one group has length p, so let $n_a = n_c = p+1$ and $n_b = p$. In this case $d = n-(p+1) = 2p+1$, which implies that up to p errors can be corrected in a word.

The rate of the bits that can be corrected with respect to the word length is $\lfloor (d-1)/2 \rfloor / n$. Substituting n and d for the three cases above gives expressions in p, which all approach $1/3$ as $n \to \infty$. So, these codes can correct asymptotically one third of a memory's bits.

As an example, we construct a code with $n = 14$, $d = 9$, and $m = 4$. Assume the first five bits of a code word belong to group a in Fig. 4, the next four bits to group b, and the last five bits to group c. Possible code words are 00000000000000, 11111111100000, 00000111111111, and 11111000011111 (see Fig. 5). This code can correct up to four arbitrary errors.

How are errors corrected under the scheme in Fig. 4? Preferrably, the correction algorithm is straightforward to keep cells simple. The most common way to correct errors in codes that are linear, which all our codes are, is by computing its syndrome S (e.g. see [13]) and deducing from it the bit positions of the errors. Required for this is the parity check matrix H of the code. For the (5,4,3)-code it is:

$$H = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Let $x = (x_1 x_2 x_3 x_4 x_5)$ be a five-bit word that may need correction. Then the syndrome is calculated as $S = Hx^{\mathrm{T}} = (x_1 + x_3 + x_4, x_1 + x_2, x_4 + x_5)^{\mathrm{T}}$. If S is the all-zero vector, no errors are detected. If there is only one bit error, its location is indicated by the value of S; we can find the location by comparing the syndrome with the columns of H. Table 1 shows the list of possible errors, the syndromes corresponding to them, and the procedures to correct them.

Table 1: Errors in a code word for the (5,4,3)-code, corresponding syndromes, and correction procedures

Errors in word $x = (x_1 x_2 x_3 x_4 x_5)$	Syndrome $S = (s_1 s_2 s_3)$	Correction Procedure
No errors	000	-
Single error in x_1	110	Flip x_1
Single error in x_2	010	Flip x_2
Single error in x_3	100	Flip x_3
Single error in x_4	101	Flip x_4
Single error in x_5	001	Flip x_5
Double errors	other	-

While the above scheme is straightforward, correcting errors by syndromes is more complicated for codes with larger distances, which may result in very complicated cells. For our (14,4,9)-code we show a different error correction method, which employs a table storing the four code words. Given a word in a memory that is to be corrected, the method calculates the distances to the table entries, and selects the code word with the smallest distance as the most likely. Called *distance comparison* [14], this method has limited complexity due to the small number of code words, and it has the added advantage that in many cases five or six errors can be corrected. For example, if there are errors in the first three bits, the sixth bit, and the tenth bit of the all-zero code word 00000000000000, it becomes 11100100010000, and this word has distance 5 to the original word, but distances 6, 10, and 7 to the code words 11111111100000, 00000111111111, and 11111000011111, respectively. As the code is linear, similar examples are possible with the other code words. In general, if there are five bit errors in a word, they can be corrected by distance comparison in the following cases.

- There are three errors in one group in the scheme in Fig. 4 and one error in the other two groups each. The number of ways to select three bits from one group and one bit from each of the other groups is:

$$C(5,3)\,C(4,1)\,C(5,1) + C(5,1)\,C(4,3)\,C(5,1) + C(5,1)\,C(4,1)\,C(5,3) = 500,$$

whereby $C(n,k) = \frac{n!}{(n-k)!k!}$ is the number of ways to select k unordered objects out of a set of n distinct objects.

- There are two errors in each of two groups and one error in the remaining group. The number of ways to select two bits from two groups and one bit from one group is:

$$C(5,2)\,C(4,2)\,C(5,1) + C(5,2)\,C(4,1)\,C(5,2) + C(5,1)\,C(4,2)\,C(5,2) = 1000.$$

There is no other way to recover five errors among 14 bits by using distance comparison. Given that the total number of ways to select five bits out of 14 bits is $C(14,5) = 2002$, we conclude that $500 + 1000 = 1500$ out of 2002 cases with five errors can be recovered by our scheme; that is about 74.9%.

Some six bit errors are also recoverable. For example, if bits 1, 2, 6, 7, 10, and 11 are erroneous in the all-zero word, we obtain 11000110011000. This word has distance 6 to the all-zero code word, and distances 7, 7, and 8 to the code words 11111111100000, 00000111111111, 11111000011111, respectively. In general, if there are six bit errors in a word, they can be corrected by distance comparison in the following case.

- There are two errors in each of the three groups in Fig. 4. The number of ways to select two bits from each group is:

$$C(5,2)\,C(4,2)\,C(5,2) = 600.$$

Given that the total number of ways to select six bits out of 14 bits is $C(14,6) = 3003$, we conclude that 600 out of 3003 cases with six erroneous bits are recoverable by our scheme; that is about 19.9%.

More in general, errors can be corrected by the distance comparison method if the number of erroneous bits in every *pair* of bit groups (among the bit groups a, b, and c in Fig. 4) does not exceed $\lfloor (d-1)/2 \rfloor$: this guarantees that every pair of bit groups is closer in distance to the corresponding pair in the original word than to the corresponding pair in any of the other words. Under this condition, up to $(3/2)\lfloor (d-1)/2 \rfloor$ errors can be corrected, and this occurs when each of the three bit groups contains at most $\lfloor (d-1)/2 \rfloor/2$ erroneous bits. So, asymptotically $(3/2)(1/3) = 1/2$ of a memory's bits can be corrected if the errors are evenly distributed over the three bit groups.

4 Conclusions and Discussion

The CA in this paper is fault-tolerant due to the representation of its cells' memory states as code words in an error correcting code. Provided efficient codes are used, this technique requires less hardware redundancy than in previous CAs. The error correction method employed compares a memory's contents to a small fixed set of correct patterns, and changes it to the pattern nearest in distance. This may resemble error correction mechanisms in nature, which tend to use templates, like in DNA replication.

Our CA allows correction even if up to about one third of the bits in each memory is corrupted. If the errors are in certain favorable positions, even more bit errors can be corrected. At best, recovery is then possible if almost half of the bits are corrupted. The ability of our coding scheme to correct errors beyond what its minimal distance theoretically allows indicates that the code words, when represented as points in hyperspace, have much space around them in selected bit positions. This phenomenon is mainly due to the limited number (four) of code words [14].

Though the proposed error correction scheme has been applied to an asynchronous CA able to conduct universal computation in [14], it may be applied to other CAs as well, like CAs on which self-reproduction [21, 22, 23] takes place. To implement self-reproduction on an asynchronous CA of the type in this paper, memories may require somewhat more bits to carry configuration information. This implies that a fault-tolerant version of the CA may require error correcting codes to have somewhat more code words than the four assumed in this paper. These codes are likely to have better efficiency than the codes in this paper, because the hyperspace will be filled more efficiently [14]. Consequently, the number of bits required to make the memories fault-tolerant may hardly have to increase to facilitate the larger number of code words.

Interactions in this paper have been assumed to be localized: state transitions of cells are based on local information and the update mechanism is locally (i.e., asynchronously) timed. Error correction is also conducted on a local scale, taking place in each memory individually. This local mode of operation seems to be compatible with interactions commonly observed in organisms. The fault-tolerant CA in this paper may thus have significance in a biological context.

References

[1] D. Mange, M. Sipper, A. Stauffer, et al (2000) "Toward robust integrated circuits: the embryonics approach," *Proc. of the IEEE*, vol. 88, no. 4, pp. 516–541.

[2] J.R. Heath, P.J. Kuekes, G.S. Snider, et al (1996) "A defect-tolerant computer architecture: Opportunities for nanotechnology," *Science*, vol. 280, pp. 1716–1721.

[3] L. Priese (1976) "On a simple combinatorial structure sufficient for syblying nontrivial self-reproduction," *J. of Cybernetics*, vol. 6, pp. 101–137.

[4] H. Nishio and Y. Kobuchi (1975) "Fault tolerant cellular spaces," *J. of Computer and System Sciences*, vol. 11, pp. 150–170.

[5] M. Harao and S. Noguchi (1975) "Fault tolerant cellular automata," *J. of Computer and System Sciences*, vol. 11, pp. 171–185.

[6] P. Gács (1986) "Reliable computation with cellular automata," *J. of Computer System Science*, vol. 32, no. 1, pp. 15–78.

[7] P. Gács (1989) "Self-correcting two-dimensional arrays," in *Advances in Computing Research (a scientific annual)*, Silvio Micali, Ed., Greenwich, Conn., vol. 5: Randomness in Computation, pp. 223–326, JAI Press.

[8] P. Gács and J. Reif (1988) "A simple three-dimensional real-time reliable cellular array," *J. of Computer and System Sciences*, vol. 36, no. 2, pp. 125–147.

[9] W. Wang (1991) *An Asynchronous Two-Dimensional Self-Correcting Cellular Automaton*, Ph.D. thesis, Boston University, Boston, MA 02215, 1990, Short version: In Proc. 32nd IEEE Symposium on the Foundations of Computer, IEEE Press, pp.188–192.

[10] P. Gács (2001) "Reliable cellular automata with self-organization," *J. of Statistical Physics*, vol. 103, no. 1/2, pp. 45–267, see also *IEEE Symp. on Foundations of Computer Science*, pp. 90–99, 1997.

[11] M.S. Capcarrere (2002) *Cellular automata and other cellular systems: design and evolution*, Ph.D. thesis, no. 2541, Swiss Federal Institute of Technology, Lausanne, Switzerland.

[12] T. Isokawa, F. Abo, F. Peper, et al (2003) "Fault-tolerant nanocomputers based on asynchronous cellular automata," *submitted.*

[13] F.J. MacWilliams and N.J. Sloane (1978) *The Theory of Error-Correcting Codes*, NorthHolland.

[14] F. Peper, J. Lee, F. Abo, et al (2004) "Fault-tolerance in nanocomputers: a cellular array approach," *IEEE Trans. on Nanotechnology*, in press.

[15] F. Peper, J. Lee, S. Adachi, et al (2003) "Laying out circuits on asynchronous cellular arrays a step towards feasible nanocomputers?," *Nanotechnology*, vol. 14, pp. 469–485.

[16] K. Nakamura (1974) "Asynchronous cellular automata and their computational ability," *Systems, Computers, Controls*, vol. 5, no. 5, pp. 58–66.

[17] F. Peper, T. Isokawa, S. Adachi, et al (2001) "Self-timing in biological systems simulated on cellular automata," in: Proc. of the 6th Int. Symp. on Artificial Life and Robotics (AROB 01), Tokyo, Japan, Masanori Sugisaka and Hiroshi Tanaka (Eds.), pp. 387–392.

[18] J. Lee, S. Adachi, F. Peper, et al (2004) "Embedding universal delay-insensitive circuits in asynchronous cellular spaces," *Fundamenta Informaticae*, in press.

[19] S. Adachi, F. Peper, and J. Lee (2004) "Computation by asynchronously updating cellular automata," *J. of Statistical Physics*, in press.

[20] J. Lee, F. Peper, S. Adachi, et al (2002) "Reversible computation in asynchronous cellular automata," in *3rd Int. Conf. on Unconventional Models of Computation 2002*, Cristian S. Calude, Michael J. Dinneen, and Ferdinand Peper (Eds.), pp. 220–229, Springer.

[21] Ed: A.W. Burks J. von Neumann (1966) *The theory of Self-Reproducing Automata*, University of Illinois Press, Champaign, IL.

[22] E.F. Codd (1968) *Cellular Automata*, Academic, New York.

[23] J.A. Reggia, S.L. Armentrout, H.-H. Chou, et al (1993) "Simple systems that exhibit self-directed replication," *Science*, vol. 259, pp. 1282–1287.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A Study of a Parallelized Immune Co-evolutionary Algorithm for Division-of-labor Problems

Naruaki Toma, Satoshi Endo, Koji Yamada
Dept. of Information Engineering
Univ. of the Ryukyus
1 senbaru, Nishihara, Okinawa 903-0213

Abstract

Recently, a number of parallelized optimization algorithms are proposed. By now, we have proposed a co-evolutionary immune algorithm (IA) to solve the division-of-labor problems, in particular n-th agent's Travelling Salesman Problem (n-TSP). In this paper, we extend the co-evolutionary IA for large-scale n-TSP with (1) the improvement for the search speed through parallelized search on the PC-cluster, and (2) the introduction of a new division-processing "pre-estimated division processing" to improve the search ability.

1 Introduction

Recently, we can get ready for large-scale computing environments such as PC-cluster as a low price. Hereby, some parallelized optimization algorithms [4, 5] are proposed: for example, parallel-distributed genetic algorithm (PDGA) and PDGA with distributed environment scheme (PDGA/DE). In PDGA, population was parallelized to improve not only the search cost but also the search ability. And PDGA/DE could the cost-saving for parameter setting.

By now, we have proposed a co-evolutionary immune algorithm (IA) to solve the division-of-labor problems, in particular n-th agent's Travelling Salesman Problem (n-TSP) [1]. The purposes of this paper are to construct a parallelized optimization system for large-scale combinatorial problems, which the system is based on an immune co-evolutionary algorithm. The co-evolutionary IA searches a best solution through the interactions between two kinds of agents, one of the agent is called immune agents, which optimize the cost of its own work. And our target problem is an n-TSP.

We extend the algorithm so as to solve large-scale problem by using parallel processing on PC cluster. Our parallelized system solves problems through the following three steps. Firstly, the target problem is divided to the sub problems depending on the area, which is a problem domain in the n-TSP. Secondary, each sub problem area is processed by the different PCs with the immune co-evolutionary algorithm. Thirdly, a main PC gathers the sub results from the other PCs, and then it adjust the optimality of the sub results as a best solution by using escape processing. To investigate the validity, we verify through some computer simulations.

2 Issues of division-of-labor problems

The key subject of division-of-labor problems is to focus on the issues of distribution of works for agents. The domain that each agent covers is defined as work domain (WD), and the aggregate total of the work domains is defined as problem domain (PD). The objective of the division-of-labor problems optimization is to find a solution, which is satisfied the following three conditions.

(a) **equaling work assignment**: In the problem domain PD, each work domain WD_i for agent$_i$ must be divided so as to have equal work assignment.

(b) **optimizing a work-cost individually**: In the divided work domain WD_i, the cost for the work of agent$_i$ must be optimized (minimized).

(c) **optimizing all amount of work-costs**: All amount of the work costs must be optimized (minimized) with satisfying (a) and (b).

Basically, the division-of-labor problems optimization is a subject, which requires minimizing a total cost in a parallel distributed processing. As the applications, it is considered as scheduling problem, job scheduling in a networked parallel computers, TSP with some

constraints[1] , vehicle routing problem (VRP) and the VRP with time constraints, and so on.

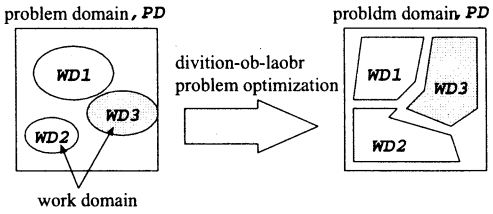


Figure 1: Division-of-labor problems optimization.

3 Proposed method

The proposed algorithm consists of the co-evolutionary IA [1] and the following extensions:

- improvement of the search speed through a parallelization of problem area
- improvement of the search ability through a introduction of a pre-estimated division-processing.

3.1 Overall procedure

In order to perform the improvement of the search speed, our algorithm solves the problem through the following procedure:

- (Stage1) separate the problem into some sub-problems based on the problem space,
- (Stage2) solve each and all sub-problem on the each node (PC) with the co-evolutionary IA, and
- (Stage3) adjust the balance among sub-solutions through the escape-processing on the main node.

Of course, to search any tours striding plural sub-problems will be prevented under the parallelization. We revel this aspect by Stage3, which gathers the sub-solutions from all nodes and adjusts the cost balance among them through the above-mentioned escape-processing.

3.2 Parallelization for problem area

Our algorithm separates the given problem (problem domain, which consists of allocations for all cities in n-TSP) and processes the separated sub-problems

[1] http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/

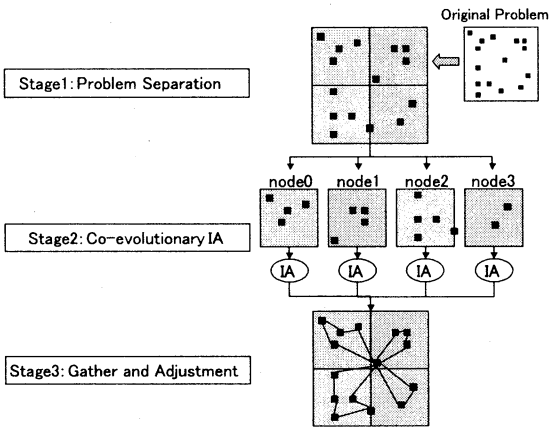


Figure 2: Global image.

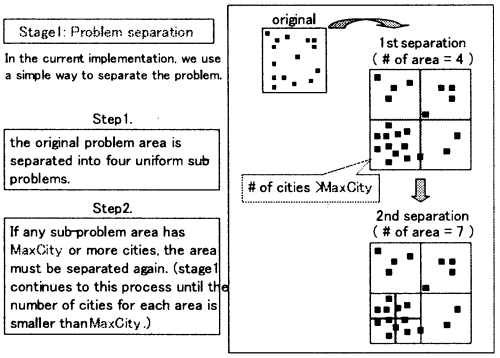


Figure 3: Problem separation.

are on each node. The problem domain separates to 2x2 matrix whose each lattice consists of same size, so there are four uniform lattices at the first stage. When any lattice has cities more than MaxCity, which is a reserved word as a limitation number for cities on node, the lattice is separated to 2x2 matrix again until the number of cities is lower then MaxCity. There are many separation methods: clustering for cities, distance based separation, and so on. However, we used a simple separation way (above-mentioned) for problem domain in n-TSP because we want to make an problem independent algorithm.

3.3 The co-evolutionary IA[1]

The co-evolutionary IA solves the problems through two searching ways, (1) division-and-integration processing by salesman agents and (2) escape processing by city agents in the environment. The procedures of

the proposed algorithm against an n-TSP are shown on figure 4.

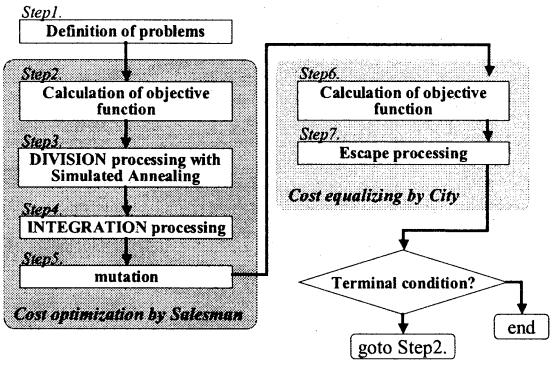


Figure 4: The co-evolutionary IA.

3.4 Pre-estimated division-processing

The objective of division-processing, which is on step 3 in the co-evolutionary IA, optimizes (decreases or minimizes) the cost through division of own tour into two subtours. For the purpose, we employed a kind of random way that decides a divided point at random and the judgment whether the division is achieved or not depends on a simulated annealing (a kind of climbing method with temperature, which the temperature denotes a possibility to go down; and the possibility is decreased step by step to converge the search). This old fashioned division-processing found better solutions, but it was a lack of a search ability for cost optimization, especially the number of cities on one salesman is over 100. In this paper, we improve this point by introduce a new way "pre-estimated division-processing" (an example shows on figure 5).

1. decides a part of own tour sub based on any two points.
   At this time, the tour consists of sub and the back-and-force two cities call Before1, and other tour consists of a remained tour except for *sub* from Before1 call After1.

2. finds the nearest city nearest from the head of sub.

3. calculates two candidate-tours' costs and finds the lower cost tour.
   One of the tour consists of the back of nearest + sub + nearest itself, and the other tour consists of nearest itself + sub + the front of nearest. From the candidates, we choose a lower cost tour as After2, and the tour consists of a remained tour expect for sub from After2 as Before2.

4. judges whether do division-processing or not depends on the estimated cost.
   We estimated that After1 + After2 as one of the optimized possibilities for better solutions, and Before1 + Before2 as a current cost. To decide the judgment to do division-processing, we compare the estimated costs as followings:

$$If Cost(A) > Cost(B) then, True \rightarrow division$$
$$False \rightarrow no\ division,$$
$$where A = Before1 + Before2,$$
$$B = After1 + After2. \quad (1)$$

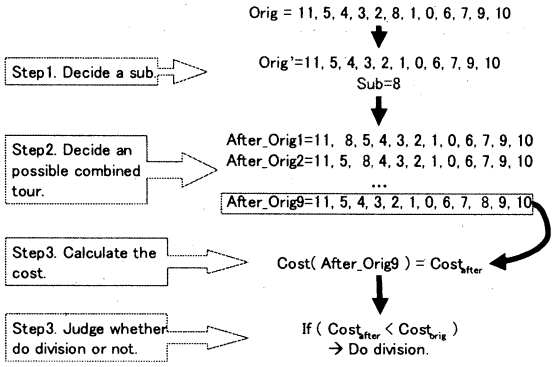


Figure 5: An example behavior of the pre-estimated division processing.

3.5 Gather and adjust the cost balance among sub-solutions

After the terminal steps for the co-evolutionary IA, the sub-solutions on each node are gathered into main node to adjust the cost balance. The adjustment is achieved by escape-processing on the main node, which has all cities at this time.

4 EXPERIMENT

4.1 Simulation design

In order to evaluate the performance of our method, we compare the results between the previous method (co-evolutionary IA *without* pre-estimated method)

and proposed method. Both algorithms are applied to two following n-TSPs: (1) dual circle 49-city n-TSP with two peaks deceptive problem and (2) 1000-city n-TSP made at random (Table 1).

Table 1: Problem settings.

	exp1	exp2
cities	49	1,000
allocation of cities	dual circle	random
start city	center	random
neighbors	5	5
min salesmen	1	3
terminal cycles	200	10,000

4.2 Results and discussions:

Table 2 shows the costs obtained by previous method (co-evolutionary IA without pre-estimated division) and proposed method. From this results, the proposed method obtained neary equal or better costs.

Table 2: (Exp1) Cost on each node.

node no.	previous	proposed
node_0	2.200276	2.053606
node_1	2.569807	2.379430
node_2	2.200277	2.148661
node_3	1.822837	1.822837

Table 3 shows a result for exp2, and the proposed method got also better costs. However, the *sum_error* denotes that the convergence ability has some wrongs, which the *sum_error* is calculated by the following formulas (i.e., the high *sum_error* has vary widely cost distribution among salesmen).

$$sum\_error = \sum_{i}^{N} |ave\_cost - x_i| \quad (2)$$

$$ave\_cost = (\sum_{i}^{N} x_i)/N \quad (3)$$

5 Conclusion

In this paper, we accelerated the search speed through a parallelization of problem area, and improved the search ability through a introduction of a pre-estimated division-processing into the co-evolutionary IA. However the current simulator was implemented up to step 2 on overall procedure, the simulation showed pretty good results except for *sum_error*. We will improve the height of *sum_error* through an implementation of step 3. In addition, we have an interest in the search ability of the pre-estimated division-processing as genetic operator on GA, the results will be shown later.

Table 3: (Exp2) Cost on each node.

node no.	previous	proposed
	(*sum_error*)	(*sum_error*)
node_0	22.804396	17.391262
	(0.250477)	(1.870297)
node_1	16.375703	12.025132
	(0.080690)	(0.014744)
node_2	14.097906	12.331801
	(0.076036)	(3.573724)
node_3	14.758087	14.672518
	(0.041723)	(0.043995)

I express the mind of appreciation to Okinawa Industry Promotion Public Corporation.

References

[1] N. Toma, S. Endo, K. Yamada, H. Miyagi: "The Immune Distributed Competitive Problem Solver Using Major Histocompatibility Complex and Immune Network". OPERATIONS RESEARCH / MANAGEMENT SCIENCE AT WORK, Kluwer's INTERNATIONAL SERIES, pp.129-147. (ISBN 0-7923-7588-2), (2000).

[2] N. Toma, S. Endo, K. Yamada: "The Proposal and Evaluation of an Adaptive Memorizing Immune Algorithm with Two Memory Mechanisms". Journal of JSAI, Vol.15, No.6, pp.1097-1106 (2002).

[3] S. Forrest, A.A. Perelson: "Genetic algorithm and the Immune system". Proc. of 1st Workshop on Parallel Problem Solving from Nature, Dortmund, Federal Republic of Germany, 1-3, October (1990).

[4] Theodore C.Belding: "The Distributed Genetic Algorithm Revised", Proc. of the Sixth International Conference on Genetic Algorithms, pp.114-121 (1995).

[5] T.Starkweather, D.Whitley and K.Mathimas: "Optimization Using Distributed Genetic Algorithms", Parallel Problem Solving from Nature, Springer Verlag, pp.176-183 (1991).

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Information Separation of Position and Direction of a Robot by two Interacting Self-Organizing Maps

Naoki OSHIRO and Koji KURATA
Faculty of Engineering, University of the Ryukyus,
Senbaru 1, Nishihara, Okinawa. 903–0213
oshiro@mibai.tec.u-ryukyu.ac.jp

Abstract

In this paper, we proposed a model to self-organize a map for a robot navigation by using visual information of itself. The robot is assumed to have visual sensors around it. The recognition model is based on Kohonen's SOM (Self-Organizing Map), which was proposed as a model of self-organization of a cortex. Ordinary SOM consists of a two-dimensional array of neuron-like feature detector units. We want to extract the information of direction and position separately from visual input, which is a function of the two information factors. Our model consists of two layers. The first layer is for directional information and consists of units arranged in a circular array, and the second layer for position information and consists of a two-dimensional array. The units in the second layer accept inputs from all the units in the first layer through plastic inhibitory synapses. It will be shown by computer simulation that the units in the first layer develop direction sensitivity and lose position sensitivity through the training, while in the second layer, the units develop position sensitivity and lose direction sensitivity.

Key Words: robot navigation, self-organization, SOM, information separation, two-factor problem

1 Introduction

We recognize our position and direction in familiar environments. It seems that we can form some kind of map in our brain from experiences. This ability is obviously desirable for robots to share. We assume a robot which has many visual sensors around it to obtain omni-directional information of the scenes, which is a function of position and direction of the robot. We propose an algorithm with which the robot can self-organize two maps to recognize the position and direction of itself as moving autonomously in a room.

In some preceding works, SOM was used for localization of a robot [2]. Also, a kind of associative neural network was used for robot navigation [8]. In this study, we focus on *separation of two factors of information.* With two maps properly organized with our model, it will be shown that the two factors can be extracted separately from the visual input.

In our previous work [5], we proposed a model for the same purpose consisting of three-dimensional unit array. The main difference of our new model and the older one is the number of the learning units. In our new model we use $N^{(1)} + N^{(2)}$ units; $N^{(1)}$ units are assigned for direction recognition and $N^{(2)}$ units for position recognition. To achieve a representation of same resolution with the older model we would need $N^{(1)} \times N^{(2)}$ units. So, we can drastically reduce the number of units with the new model.

2 Kohonen's SOM Algorithm

SOM (Self-Organizing Map) was proposed by Kohonen [3, 4] as a model of cerebral cortex and its self-organization. It was successful for SOM to reproduce the functional map on visual cortex [1, 7], and was applied to many kinds of data as a statistical tool of nonlinear auto-regression [4].

SOM usually consists of the two-dimensional array of neuron-like units, each of which has an *reference vector* $\boldsymbol{m}_i$, which is n-dimensional as well as input vectors $\boldsymbol{x}$.

SOM algorithm is as follows:

(SOM1) Assign random values for the reference vectors.

(SOM2) Choose or generate an input vector $\boldsymbol{x}$ in some random manner.

(SOM3) Find the unit whose reference vector is the closest to the input vector $\boldsymbol{x}$ in the sense of n-dimensional Euclidean distance. The unit is called the *winner.*

(SOM4) Apply the following learning rule for the reference vectors:

$$\boldsymbol{m}_i := \boldsymbol{m}_i + h_{ci}(\boldsymbol{x} - \boldsymbol{m}_i), \tag{1}$$

$$h_{ci} = \alpha \exp\left(-\frac{\|\boldsymbol{r}_c - \boldsymbol{r}_i\|^2}{2\sigma^2}\right), \tag{2}$$

where $\boldsymbol{r}_i$ is the two-dimensional position of the unit i on the unit array, and c is the index assigned to the winner unit.

(SOM5) Return to (SOM2) and repeat (SOM2) - (SOM4) many times.

This is called *neighborhood learning*, because a modification of the reference vectors takes place mainly in the neighborhood of the winner, as is represented by h_{ci}, which is called the *neighborhood function.* As the result of this learning rule, the reference vectors are scattered all over the input signal region and neighboring units tend to grow similar reference vectors, which means a topological map of the input signal region is *self-organized* on the unit array.

3 Input Data

Input data is obtained from visual sensors of the robot. The robot is placed in a square room with four walls with four different colors for each.

The robot has 80 sensors all around it, each of which can distinguish the color of the wall in front of it (Figure 1). The information obtained by one sensor is represented by one of the following four-dimensional vectors, (1,0,0,0), (0,1,0,0), (0,0,1,0), (0,0,0,1), so that the whole visual input consists of 80 four-dimensional vectors. The vectors are divided into four sets and summed up within each set to reduce the number of vectors. Then the resulting four vectors are connected to be a 16-dimensional input vector $\boldsymbol{x}$. The reference vectors of the units are also 16-dimensional.

4 Separation of Direction and Position

The first layer of our model is designed to extract the information of the direction of the robot. Let us call this layer the *direction layer.* Because the direction has the topology of a circle, the first layer is one-dimensional and the units are arrayed along a circle. On the other hand, the second layer is for the position in a room. Therefore, the units on the second layer are arrayed two-dimensionally. Let us call the second layer the *position layer.*

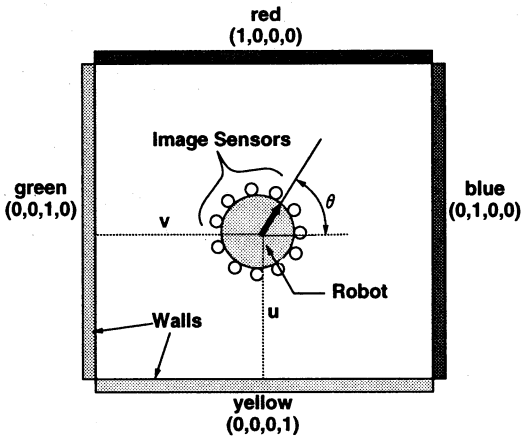


Figure 1: Robot in a room and sensors all around it. The robot walks around in the room autonomously and each sensor obtains color information of the walls.

Each layer recieves the same visual input data (Figure 2). The input changes more drastically depending on direction than position. The first layer, therefore, extracts direction information if parameters are set adequately. We assume plastic inhibitory connections from all the units in the first layer to all the units in the second layer. It is the role of these connections to force the second layer to extract some information independent of that extracted by the first layer.

The robot moves randomly within the room. Here is an important assumption that the robot changes its direction in same position and learn only if the winner of the position layer is unchanged.

The learning rule is detailed as follows:

(RN1) Assign random values for all reference vectors in the both layers $\boldsymbol{m}_i^{(1)}, \boldsymbol{m}_j^{(2)}$, where $i = 1, ..., N^{(1)}$, $j = 1, ..., N^{(2)}$. Initialize inhibit connections $s_{j,i}$ to zero.

(RN2) Set the position $\boldsymbol{u} = (u, v)$ and two directions θ, ϕ of the robot randomly. Calculate two input vectors $\boldsymbol{x}(\boldsymbol{u}, \theta)$, $\boldsymbol{x}(\boldsymbol{u}, \phi)$.

(RN3) Find the winners, $c_\theta^{(1)}$ and $c_\phi^{(1)}$ in the first layer for the two inputs $\boldsymbol{x}(\boldsymbol{u}, \theta)$, $\boldsymbol{x}(\boldsymbol{u}, \phi)$, respec-

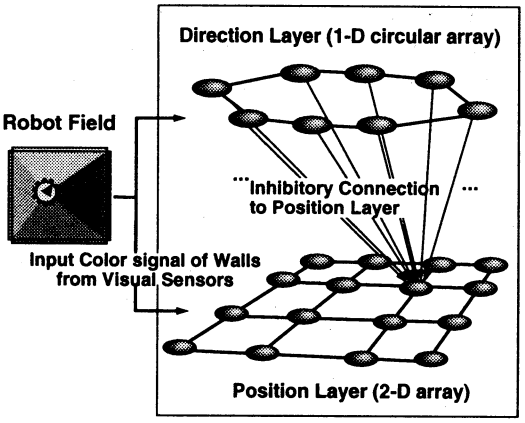


Figure 2: Structure of proposed model. Layers are designed for information separation. Visual signal of colored walls given to both layers.

tively.

$$c^{(1)} = \underset{i}{\operatorname{argmin}} \left|\left| \boldsymbol{m}_i^{(1)} - \boldsymbol{x} \right|\right|. \tag{3}$$

Then find the winners $c_\theta^{(2)}$ and $c_\phi^{(2)}$ in the second layer for the two inputs, respectively, considering the inhibitory input from the winner in the first layer.

$$c^{(2)} = \underset{j}{\operatorname{argmin}} \left\{ \left|\left| \boldsymbol{m}_j^{(2)} - \boldsymbol{x} \right|\right|^2 - s_{c^{(1)},j} \right\}. \tag{4}$$

Apply the learning rule (RN4) only if the two winners of the second layer coincides, $c_\theta^{(2)} = c_\phi^{(2)}$. Otherwise, skip (RN4) and jump to (RN5).

(RN4) Update the reference vectors of all the units in both layers,

$$\boldsymbol{m}_i^{(l)} := \boldsymbol{m}_i^{(l)} + h_{c^{(l)}i} \left(\boldsymbol{x} - \boldsymbol{m}_i^{(l)} \right), \quad (l = 1, 2). \tag{5}$$

(RN5) Update the inhibitory connections,

$$s_{j,i} := s_{j,i} + \Delta s_{j,i}, \tag{6}$$

$$\Delta s_{j,i} = \begin{cases} \left(1 - 1/N^{(2)}\right)\kappa, & i = c^{(1)},\ j = c^{(2)}, \\ -\kappa / N^{(2)}, & \text{otherwise.} \end{cases} \tag{7}$$

(RN6) Return to (RN2) and repeat (RN2) $\sim$ (RN5) many times.

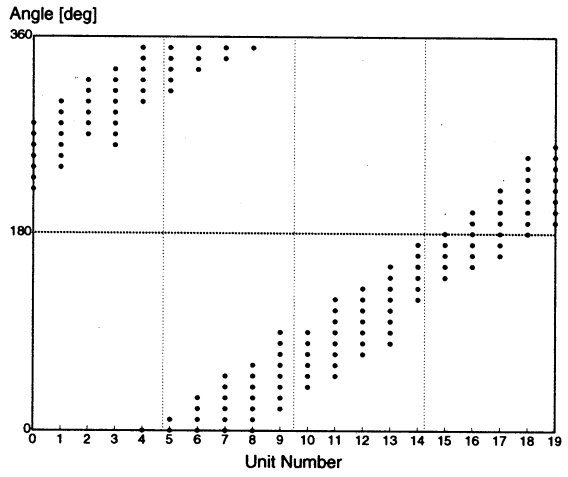
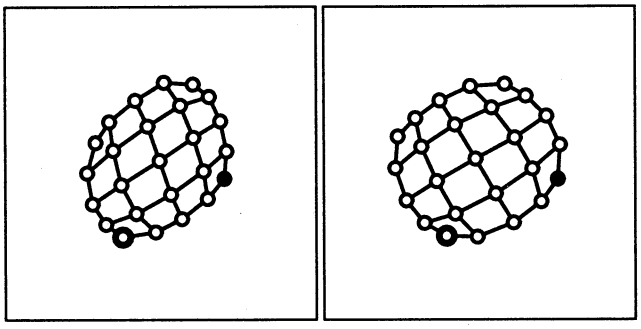


Figure 3: Prefered directions of the first layer units in various positions.

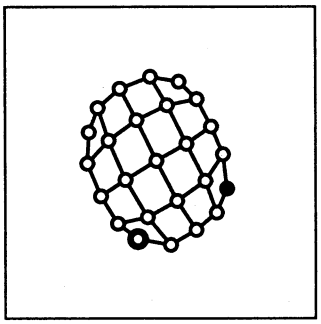
5 Simulation Results

We used $N^{(1)} = 20$ and $N^{(2)} = 5 \times 5$ units for the two layers, respectively. Parameters were set as follows. Total leaning steps $t_{\max} = 300,000$. κ in equation (7) was set to 0.1. For the first layer, learning factor $\alpha^{(1)} = 0.005\ (t < 30,000), 0.001\ (t \geq 30,000)$, the width of the neighborhood function $\sigma^{(1)} = 17e^{-t/10,000}$). Here, t denotes the descrete learning step number. For the second layer $\alpha^{(2)} = 0.001\ (t < 40,000), 0.0001\ (t \geq 40,000)$, and $\sigma^{(2)} = 17e^{-t/30,000}$ as far as this is larger than 0.5. We assumed $\sigma^{(2)}$ would not go less than 0.5.

Learning results of the direction and location layers are shown in Figure 3 and 4, respectively. After the leaning, each of the units acquires its *preferred position* and *preferred direction*. In Figure 3, the prefered direction of each cell in the first layer in various positions in the room is plotted. We can see the prefered directions are hardly influenced by positional change. In Figure 4, the prefered position of each cell in the second layer is shown. The prefered positions are averaged over three different ranges of angles $0 \sim 120°$, $120 \sim 240°$, $240 \sim 360°$, and we have obtained three figures roughly alike. This means that in the second layer the prefered direction of each cell is hardly influenced by directional change. These results mean that the information is separated by our algorithm, and the two kinds of information are mapped onto the two different layers.



(a) $0 \sim 120°$ (b) $120 \sim 240°$



(c) $240 \sim 360°$

Figure 4: Prefered positions of the second layer units averaged over three different ranges of direction.

6 Discussion

After the self-organization of the maps, the winner of the first layer tells us the direction of the robot in the room, and the winner of the second layer the direction. Uniform winning rate of the second layer is given by the plastic inhibitory connections from the first layer. These connections help the two layers to extract two features distributing independently of each other. This algorithm may not be successfully applied to a robot in a room whose shape is unknown, as well as the previously proposed method [5], because the second layer in our model are rigid square-shaped array. Our model could be improved by adopting *neural gas* algorithm for the unit layers. Neural gas, proposed by Martinetz [6], is more flexible and can fit for wide variety of shapes, topologies, dimensions.

7 Conclusion

We have proposed a self-organizing model which can learn to recognize from visual input the position and direction of a robot in a room. During the learning phase, the robot is assumed to move randomly within the room. The model consists of two SOM layers connected through plastic inhibitory synapses. Position and direction are mapped separately onto the two layers.

References

[1] Erwin, E., Obermayer, K., Schulten, K. (1995). Models of Orientation and Ocular Dominance Columns in the Visual Cortex: A Critical Comparison. *Neural Computation*, **7**, pp.425-468.

[2] Gerecke, U., Sharkey, N. (1999). Quick and Dirty Localization for a Lost Robot. In *Proceedings of IEEE International Conference on Computational Intelligence for Robotics and Automation (CIRA-99)*, Monterey, CA.

[3] Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Map. *Biological Cybernetics* **43**, pp.59-69.

[4] Kohonen, T. (1994). *Self-Organizing Maps*, Springer-Verlag, Berlin.

[5] Kurata, K., Oshiro, N. (2002). Information Separation of Position and Direction of a Robot by Self-Organizing Map. *In Proc. of 8th AROB*, **2**, pp.641-644.

[6] Martinetz, T., Berkovich, S., Schulten, K. (1993). Neural-Gas Network for Vector Quantization and its Application to Time-Series Prediction. *IEEE Trans. on Neural Networks*, **4**, pp.558-559.

[7] Shouno, H., Kurata, K. (2001). Formation of a Direction Map using Kohonen's Self-Organization Map by Projection Learning. *Biological Cybernetics*, **85**, pp.241-246.

[8] Mizutani, K., Omori, T. (1999). On-line Map Formation and Path Planning for Mobile Robot by Associative Memory with Controllable Attention, In *Proc. of IJCNN'99*

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

System Identification Using Dynamical Neural Network with GA-based Training

Kunihiko Nakazono
University of the Ryukyus
Senbaru 1, Nishihara, Okinawa. 903–0213
nakazono@tec.u-ryukyu.ac.jp

Kouhei Ohnishi
Keio University
Hiyoshi 3–14–1, Yokohama. 222–8522
ohnishi@sd.keio.ac.jp

Hiroshi Kinjo
University of the Ryukyus
Senbaru 1, Nishihara, Okinawa. 903–0213
kinjo@tec.u-ryukyu.ac.jp

Abstract

In this paper, we propose a dynamical neural network (DNN) having the properties of inertia, viscosity, and stiffness and its training algorithm based on a genetic algorithm (GA). In a previous study, we proposed a modified training algorithm for the DNN based on error backpropagation. However, in the former method it was necessary to determine the values of the DNN parameters by trial and error. In the proposed DNN, the GA is designed to train not only the connecting weights but also the parameters of the DNN. Simulation results show that the DNN trained by GA obtains good training performance for time series patterns generated from sine functions.

1 Introduction

Recently, recurrent neural networks and spiking neural networks have attracted more research interest than layered neural networks having static mapping capability [1, 2, 3, 4]. The recurrent neural network is a possible candidate for improving the system dynamics because it incorporates a feedback structure in the neuron unit and takes time delayed inputs into consideration. Research on spiking neural networks is also ongoing. Spiking neural networks treat spike trains and process the signals based on spike pulses. However, the network structure in recurrent neural networks and spiking neural networks is complex compared to that in layered neural networks with a training algorithm.

Here, we propose a dynamical neural network (DNN) that realizes a dynamic system and has a network structure with the properties of inertia, viscosity, and stiffness without time delayed inputs. In a previous study, the proposed DNN was constructed with a training algorithm that used error backpropagation [5]. However, that algorithm modified only the connecting weights and the property parameters for the DNN had to be determined by trial and error. We designed a GA-based training [6] both the connecting weights and the parameters of the DNN.

The validity of the proposed DNN was verified by identifying periodic functions such as a simple one-period sine waveform and several periodic sine waveforms. Simulation results showed that the proposed DNN provides higher performance than the conventional neural network.

2 Structure of DNN

In this paper, a DNN is configured using a neuron having the properties of inertia, viscosity, and stiffness. In this model, we assume the image output from neuron possesses the properties of inertia, viscosity, and stiffness, and that the output is propagated in the next neuron. The proposed DNN is composed of three hierarchy layers and the proposed neuron adopts a hidden layer and an output layer. The structure of the DNN is shown in Figure 1.

The equations for the DNN are expressed as follows.

$$y_i = u_i, \quad (i = 1, \cdots, N_I) \tag{1}$$

$$y_j = K_j f_j(net_j) + D_j \dot{f}_j(net_j) + M_j \ddot{f}_j(net_j) \tag{2}$$

$$net_j = \sum_{i=1}^{N_I} w_{ij} y_i, \quad (j = 1, \cdots, N_J) \tag{3}$$

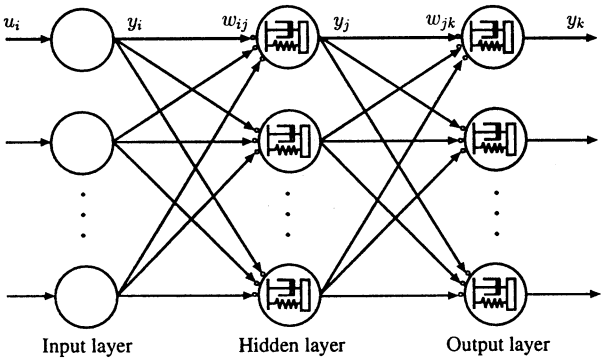


Figure 1: Structure of DNN

$$y_k = K_k f_k(net_k) + D_k \dot{f}_k(net_k) + M_k \ddot{f}_k(net_k) \quad (4)$$

$$net_k = \sum_{j=1}^{N_J} w_{jk} y_j, \quad (k = 1, \cdots, N_K) \quad (5)$$

Here, u shows input to the DNN, and y shows output from each neuron. net is the total sum of products of the connecting weight w and the output y from each neuron. M, D, and K are the property parameters of inertia, viscosity, and stiffness, respectively. The subscripts i, j, and k are the number of the input layers, hidden layers, and output layers, respectively. N_I, N_J, and N_K are the number of neurons in the input layers, hidden layers, and output layers, respectively. $f(\cdot)$ is a sigmoid function in the range of $[-1, 1]$.

3 Training algorithm based on GA

The DNN is trained using a GA in an off-line process. Figure 2 shows the flowchart of the evolution process in the DNN. The evolution algorithm for the DNN is as follows.

STEP1: Produce the initial DNNs at random. The connecting weights (w_{ij}, w_{jk}) and the property parameters (M_j, D_j, K_j, M_k, D_k, K_k) of DNNs are transformed to the chromosome.

STEP2: Sum all of the fitnesses for the DNNs.

STEP3: Select the parent DNNs by means of roulette wheel parent selection.

STEP4: Perform a crossover operation for the chromosome to produce new DNNs.

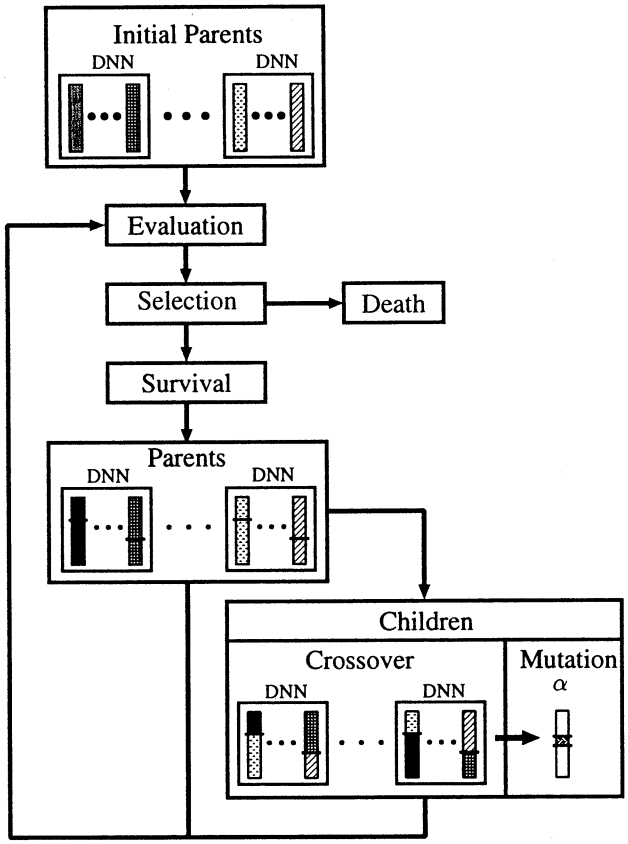


Figure 2: Process of GA-based training

STEP5: Perform a mutation operation for some additional new DNNs.

STEP6: Sum all of the fitnesses for the DNNs including the new DNNs. Go back to STEP2 until the evolution process arrives at generation 100,000.

Further information regarding the parameters of the GA is shown in Table 1.

Table 1: Simulation parameters of GA

Initial DNNs	400 individuals
Selection pressure	Roulette wheel parent selection 0.6
Crossover	One-point crossover
Mutation rate α	Bit mutation 0.10, 0.15, 0.20, 0.25, 0.30

During the GA-based training process, an error function E is used to evaluate the performance of each

DNN. E is described by the following equation as

$$E = \frac{1}{2}\sum_k e(t)^2 = \frac{1}{2}\sum_k (d(t) - y(t))^2 \qquad (6)$$

where $d(t)$ is the desired signal. The fitness of DNN is expressed in terms of the inverse of the error function E. The connecting weights and property parameters of the DNN are modified in order to maximize the fitness determined by the error function in (6).

4 Simulation

The effectiveness of the DNN proposed in this paper is verified by numerical simulation of the ability of the system to identify a periodic function. The DNN is structured to have a single input and single output (SISO). The method by which a time series signal can be identified is shown in Figure 3. The desired signal,

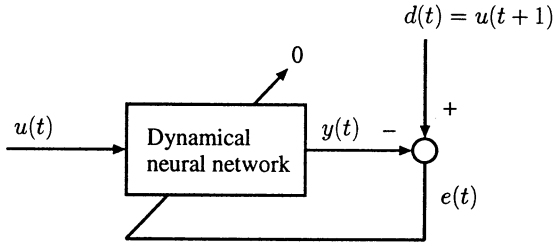


Figure 3: System identification of periodic function

namely the training data $d(t)$, is the signal that has passed one sampling period prior to the input signal $u(t)$.

4.1 Simulation 1 (single periodic function)

In order to facilitate analysis, simulation shows that the DNN identified the time series signal of the single sine periodic function with cycle T as

$$u(t) = \sin\left(\frac{2\pi t}{T}\right). \qquad (7)$$

In this case, the number of neurons in the hidden layer is five, and cycle T equals 12.

The evolution processes of the GA is shown in Figure 4. The values for the acquired property parameters of the DNN are shown in Table 2 as the evolution process approaches generation 100,000.

The results of the evolution processes show that the DNN with a mutation rate $\alpha = 0.15$ provides the best performance. All of the fitness values increase gradually and the evolution almost stagnates at generation 60,000. The result of regenerating the signal using the trained DNN is shown in Figure 5. The output of the DNN deviates negligibly from the desired signal.

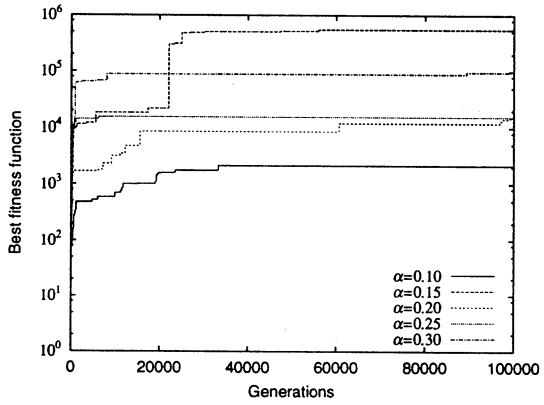


Figure 4: Evolution process (simulation 1)

Table 2: Property parameters of the DNN (simulation 1, $\alpha = 0.15$)

j	1	2	3	4	5
M_j	0.0002	9.8852	6.2499	8.7517	5.1829
D_j	4.6033	6.4100	2.5652	4.5367	4.5583
K_j	4.3355	1.8807	8.9511	8.6836	3.1583
k	1				
M_k	1.3817				
D_k	4.5214				
K_k	2.4997				

4.2 Simulation 2 (Several periodic functions)

In this section, the simulation shows that a signal having more than one cycle can be correctly identified. The number of neurons in the hidden layer is five and the number of cycles T_p is either 8, 12 or 16.

The evolution processes of GA is shown in Figure 6. The values acquired for the parameters in the DNN are shown in Table 3. The result of the evolution processes shows that a mutation rate of $\alpha = 0.30$ provides the best performance for the DNN. In simulation 2, the time course of some fitness values improved at higher mutation rates. The results of regenerating the signal using the trained DNN are shown in Figure 7. The outputs of the DNN deviated negligibly from the desired signal.

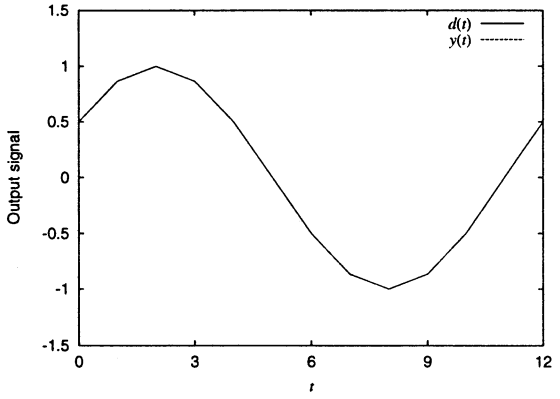


Figure 5: Regenerated waveform($\alpha = 0.15$)

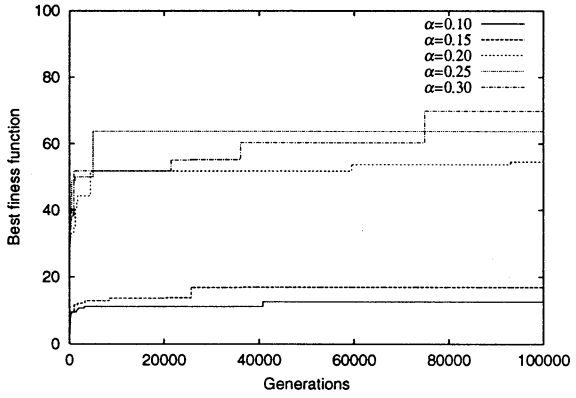


Figure 6: Evolution processes (simulation 2)

Table 3: Property parameters of the DNN (simulation 2, $\alpha = 0.30$)

j	1	2	3	4	5
M_j	2.6111	7.3390	7.2564	2.5905	0.5957
D_j	0.6981	8.6354	5.0730	1.7308	2.7683
K_j	8.4735	8.8145	9.1591	8.0311	6.6468
k	1				
M_k	1.2444				
D_k	0.1598				
K_k	2.3563				

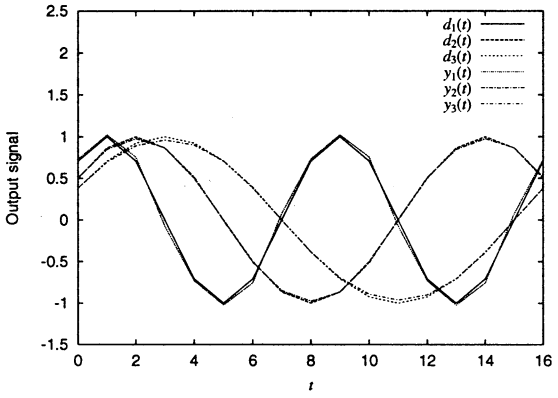


Figure 7: Regenerated waveforms($\alpha = 0.30$)

5 Conclusion

In this paper, a DNN, exhibiting the effectiveness of a neuron and capable of passing on properties of inertia, viscosity, and stiffness, was configured. Simulation results showed that the DNN trained by the GA realized good training performance for time series patterns generated from sine functions.

References

[1] D. E. Rumelhart, J. L. McClelland, and the PDP Research Group (1989), Parallel Distributed Processing, *The MIT Press.*

[2] R. J. Williams and D. Zipser (1989), A Learning Algorithm for Continually Running Fully Recurrent Neural Networks, *Neural Computation*, 1, No.2, pp.270–280.

[3] H. Kinjo, K. Nakazono, and T. Yamamoto (1997), Pattern Recognition for Time Series Signals Using Recurrent Neural Networks by Genetic Algorithms (in Japanese), *Trans. of ISCIE*, Vol. 10, No. 6, pp.304–314.

[4] W. Mass and C. M. Bishop (1999), Pulsed Neural Networks, pp. 16-53, *MIT Press.*

[5] K. Nakazono, K. Ohnishi, H. Kinjo, and T. Yamamoto (2003), Identification of Periodic Function Using Dynamical Neural Network, *AROB 8th '03*, Vol.2, pp. 633–636.

[6] Edited by L. Davis (1991), Handbook of Genetic Algorithms, *Van Nostrand Reinhold.*

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Tracking and Recognition of Moving Objects in a Monocular Image Sequence Based on Visual Characteristics

Satoru ODO†,‡ Kiyoshi HOSHINO§ Koji YAMADA†
E-mail: odo@bme.ie.u-ryukyu.ac.jp

†Department of information engineering, University of the Ryukyus
1 Senbaru, Nishihara-cho, Okinawa 903–0213 Japan
‡Department of Welfare and Culture, Okinawa University
555 Kokuba, Naha, Okinawa 902-8521 Japan
§Institute of Engineering Mechanics and Systems University of Tsukuba
Tsukuba, Ibaraki 305–8573 Japan

Abstract

The more friendly communication can be promoted between the human and computer if the function of gesture recognition is implemented to the computer system as the input interface along with the keyboards and mice. We propose a mouse-like function for estimating hand shape from input images with a monocular camera, with which a computer user feels no restraint or awkwardness. Our system involves conversion of sequential images from Cartesian coordinates to log-polar coordinates. Temporal and spatial subtractions and color information are used to extract the hand region. The original of log-polar coordinates is chosen as the center of the acquired image, but once the hand has been extracted, the estimated centroid position of the hand region in the next frame, obtained from the current hand position and speed, is used as the origin to convert. Recognition of the hand shape is carried out by the competitive neural network (NN) using higher order local autocorrelation features of log-polar coordinate space. Mouse-like functions are realized with the hand shape and motion trajectory. Compared to conventional Cartesian coordinates, conversion to log-polar coordinates enables us to reduce image date and computation time, remove the variability by the scaling, and improve antinoise characteristics.

Keyword: monocular camera, log-polar mapping, higher order local autocorrelation feature, learning vector quantization

1 Introduction

A lot of methods of object tracking and recognition has been proposed in the past[1, 2]. Some types of motion captures are available with contact sensors: such as data glove, magnetic, accelerometers, and angle sensors for gesture recognition. In those systems, attaching the sensors to the human body and using the motion capture apparatus lead high-speed, stable processing with high accuracy. However, those systems require computer users to put on special sensors restricting their motions and making them awkward, which may make the users feel uncomfortable.On the other hand, there is another type of motion capture using non-contact sensors: optical system for gesture recognition. A computer user feels free to move because of no special sensors attached to his body. However, usual optical motion capture systems need enormous computational complexity to estimate human motion indirectly. It is therefore difficult to accomplish real-time processing without using extra hardware for speed-up processing.

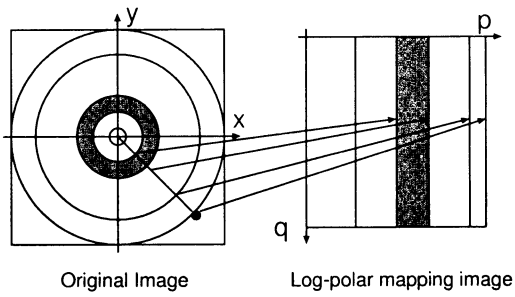
The goal of this study construct a human-computer communication system. In this paper, we propose a method to realize mouse-like functions by estimating human hand gestures from images acquired through a monocular camera. Our system involves the conversion of sequential images from Cartesian coordinates to log-polar coordinates. Temporal and spatial subtractions and color information are then used to extract the hand region. At the outset, the origin of log-polar coordinates is chosen as the center of the acquired image, but once the hand region has been extracted, the estimated centroid position of the hand region in the next frame, obtained from the current hand position and speed, is used as the origin to convert. The hand shape is identified by the NN using higher order local autocorrelation features of log-polar coordinate space. Mouse-like functions are realized by

implementing gesture recognition of hand shape and motion trajectory. The central part has high resolution, and the circumference part has rough resolution by the logarithmic sampling in the radial direction. Therefore, this mapping provides the reduction of the date size without accuracy decreased near the target. In addition, this mapping provides an invariant representation of the targets, because rotations and scalings on Cartesian coordinates are transformed into translations on log-polar coordinates.

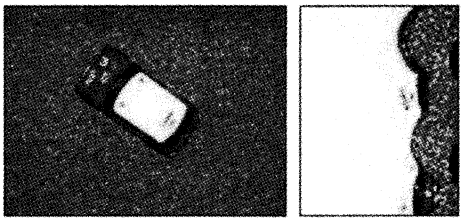
2 System

2.1 Gesture recognition algorithm

Sequential images are recorded with a monocular camera, and then converted to log-polar coordinate images. Our system generates images using log polar mapping (LPM). The advantages of LPM are that high resolution and a wide working field are obtained using relatively few pixels, while scaling invariance and rotational invariance are enabled for the center of conversion. Less image data reduces the computation time required for image processing. In our system, a contour image is generated from the LPM image using temporal and spatial subtractions and skin color information, after which the centroid of the contour image is used to estimate the position of the hand region. The destination of the hand region is estimated from data on past positions. For recognition of the hand shape, higher order local autocorrelation features are computed from the LPM image, followed by the NN. Because the translation distance is computed without complex computations such as for chirp transforms, processing is speeded up. The position is thus estimated from the centroid, from which detailed information on shape has been omitted, so there is less such likelihood of poor tracking precision caused by drastic changes in the image. Color information extracted from the skin color region and the background difference is used to eliminate background objects with similar color information, thus enabling the target to be extracted properly. We considered what kind of movement to be recognized for implementing the function of gesture recognition to the computer system as the input interface along with the keyboards and mice. As a result, we conducted recognition of intentional movements rather than that of arbitrary movements for avoiding erroneous control information, which might be generated by no intentional movements. Therefore, we decided that a user turns on/off the mouse-like function by making a predefined motion "enter"



(a) Log-polar mapping from the cartesian plane to the log-polar plane.



(b) Original image (320×240 pixels).

(c) Log-polar mapped image (60×80 pixels).

Figure 1: Constructing the log-polar mapped image.

in front of the camera.

2.2 Log polar mapping

Coordinates, $I(x, y, t)$, of the Cartesian image at time t is assumed to form complex plane Z. A point on this complex plane is expressed by $z = x + iy$. Similarly, coordinate $L(p, q, t)$, of the LPM is assumed to form complex plane W, a point of which is expressed by $w = p + iq$. LPM is shown as follows :

$$w = \log(z + \alpha) \tag{1}$$

where α is an offset to prevent singularity at the origin.

The object is focused and recognized with high resolution in the central vision, but the area is small. However, the objects in a large area are roughly recognized in peripheral vision with low resolution. An example of transformation is shown in figure 1. (a) is log-polar mapping from the cartesian plane to the log-polar plane. (b) is original image. (c) is an example of log-polar mapped image.

2.3 Estimate of position

Temporal and spatial subtractions and skin color information are used to extract the hand region and

to track it. The temporal subtraction is used to separate background and movement, since movement in sequential images corresponds to observed changes in luminance over time. Being dependent on the difference in luminance between the background and movement region, the temporal subtraction is also easily affected by changes in room illumination. Thus, edge information obtained from spatial subtractions, which are little affected by room illumination, is used so areas of the image where great luminance changes occur lie near the edge of the movement region. Skin color information is used because our system ' s purpose is to extract the hand region. Regarding how the hand-finger region is extracted from the input image, denoted by $L(p,q,t)$ the LPM image generated from the input image at time t, the image obtained by taking the temporal subtraction of $L(p,q,t)$, denoted by L_1(p,q,t), that was obtained by taking the spatial subtraction, denoted by L_2(p,q,t), and the skin color region, denoted by L_3(p,q,t), are computed as shown below.

Let $L(p,q,t)$ denote the pixel value at a point (p,q) in the log-polar mapped image. A frame subtraction image $L_1(p,q,t)$ is calculated by frame subtraction as follows:

$$L_1(p,q,t) = \begin{cases} 1 \ , & |L(p,q,t) - L(p,q,t-1)| \geq th_t \\ 0 \ , & otherwise \end{cases} \quad (2)$$

An edge image $L_2(p,q,t)$ is calculated by Sobel operator as follows:

$$L_2(p,q,t) = \begin{cases} 1 \ , & \sqrt{L_{HS}(p,q,t)^2 + L_{VS}(p,q,t)^2} \geq th_s \\ 0 \ , & otherwise \end{cases} \quad (3)$$

Parameters in RGB color coordinate system are easily influenced by brightness baseline of the image.

Therefore, in this paper, $L^*u^*v^*$color coordinate system is adopted in our system in spite of RGB color system, and then we use the u^*v^* plane in $L^*u^*v^*$color coordinate system.

A skin color image $L_3(p,q,t)$ is calculated as follows:

$$L_3(p,q,t) = \begin{cases} 1 \ , & (L_c - M)^T C^{-1} (L_c - M) \geq th_c \\ 0 \ , & otherwise \end{cases} \quad (4)$$

where M is the mean vector of skin color and C is the variance-covariance matrix of skin color.

A motion edge image $L_d(p,q,t)$ is calculated as follows:

$$L_d(p,q,t) = \begin{cases} 1 \ , & \sum_{i=1}^{3} L_i(p,q,t) = 3 \\ 0 \ , & otherwise \end{cases} \quad (5)$$

The human hand displacement $p_{xy}(t)$ is determined by the center of the motion edge image.

2.4 Estimation of motion

For tracking for the position of the centroid of the hand region, we denote the centroid position of the hand region at time t by $p_{xy}(t)$, displacement velocity of the hand region by $v(t)$, acceleration by $a(t)$, and the centroid position at time t estimated from that at time $t-1$ by $p_{xy}(t)$. Initially at $t=0$, the actual and estimated centroid position of the hand region are both set at the center of the captured image, with velocity and acceleration both assumed to be zero. Estimated centroid position $p_{xy}(t)$ at time t is given by equation (6). This then becomes the center of the log polar coordinate space for mapping the Cartesian image onto an LPM image.

$$\hat{p}_{xy}(t) = p_{xy}(t-1) + v(t-1) \cdot \Delta t \quad (6)$$

where Δt is the frame interval.

When centroid position $p_{xy}(t)$ of the actual hand region does not coincide with the estimated centroid position, $p_{xy}(t)$, we assume that acceleration, given by equation (7), took place, so displacement velocity $v(t)$ is given by equation (8).

$$a(t-1) = \frac{2}{(\Delta t)^2} (\hat{p}_{xy}(t) - p_{xy}(t)) \quad (7)$$

$$v(t) = v(t-1) + a(t-1) \cdot \Delta t \quad (8)$$

2.5 Estimate of shape

To realize mouse-like functions, the system must distinguish between "pointing" the left and right mouse buttons. Together with the hand shape used as the "enter" key, the system must be able to recognize at least four classes. For the present system, the four hand shapes are used. Higher order local autocorrelation features[3] of edge image L_2 were used to recognize hand shape. Because these features have the advantage of being translation-invariant, their extraction from the LPM image yields features that are invariant to rotation and scaling. Hand shape features are extracted from edge image L_2, which contains the

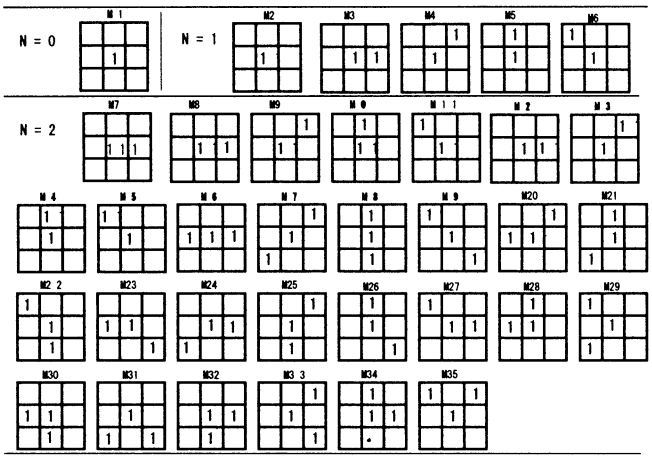


Figure 2: Local mask patterns for computing higher order local autocorrelation features.

shape of the entire hand and fingers, and not the skin color region because, in the latter, parts of the hand region may become lost by shadow effects due to the hand ' s position in relation to illumination. Denoting the target image by $I(r)$, the N-th order autocorrelation function is defined for displacement directions $(a_1, a_2, \ldots, a_N)$ as follows:

$$x^N(a_1, a_2, \ldots, a_N) = \int I(r)I(r+a_1)\ldots I(r+a_N)dr \qquad (9)$$

For this case, we used higher order autocorrelation features of order 2 (N=0,1,2), with displacement directions limited to a local region of 3 × 3 pixels centered at reference point r. Eliminating features that remain equivalent in translation, we obtain the 35 features in figure 2, where '1' represent corresponding pixels in the local pattern. The '11' and '111' in the mask patterns represent the product and cube of the same pixel value. Each feature is computed by adding to all pixels the product of corresponding pixels in the local pattern. In the pattern recognition, NN have been adopted. Among them a competitve NN has a simple structure and can explain the input/output relation more easily than a layered NN based on the back-propagation algorithm. Thus, we adopt a competitive NN for the estimate of shape where the optimized-learning-rate learning vector quantization 1 (OLVQ1)[4] is used for training.

3 Experiment and evaluation

We performed experiment to evaluate the performance of the hand tracking and recognition using Higher order local autocorrelation features and OLVQ1. Four users were asked to make the four hand postures. Captured images were transmitted to the PC at a 360 × 240 resolution via IEEE1394. Results of hand shape recognition were displayed in the shape of a mouse cursor, which notified the user of recognition results. This enables the user to modify his/her hand shape so recognition takes place correctly. This experiment showed the average processing time was 30 ms per frame, satisfactory for practical use.

4 Conclusions

We proposed tracking and recognition method from input images with a monocular camera, which as a non-contact sensor does not impart to the user a feeling of restraint or awkwardness. The sequential image is converted from Cartesian coordinates to log-polar coordinates, and temporal and spatial subtractions and color information are used to extract the hand region. The hand shape is identified by the OLVQ1 using higher order local autocorrelation features in log-polar coordinate space. Mouse-like functions are achieved by recognizing the hand shape and tracking hand movement. In our experiment, we detected hand shapes to achieve mouse-like functions stably with processing time of about 30 ms per frame. The experimental result show the effectiveness of the proposed method.

References

[1] Liang Wang, Weiming Hu, Tieniu Tan (2003), Recent developments in human motion analysis. pattern recognition, vol.36, pp.585–601

[2] Irena Koprinska, Sergio Carrato (2001), Temporal video segmentation: A survey. signal processing image communication, vol.16, pp.477-500

[3] N. Otsu and T. Kurita (1988), A new scheme for practical, fiexible and inteligent vision systems. Proceedings of IAPR Workshop on Computer Vision, pp.431–435

[4] Teuvo Kohonen (1995), Self-Organizing Maps. Springer Series in Information Sciences, vol.30

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Quantitative analysis of relationship between leg movements and subjective impressions in CG animation

Shihoko Kamisato* Satoru Odo* Koji Yamada* Shiro Tamaki* Kiyoshi Hoshino**

* Faculty of Engineering
Ryukyu Univ.
Okinawa, Japan, 903-0213

**Institute of Engineering Mechanics and Systems
Tsukuba Univ.
Ibaragi, Japan, 305-8573

Abstract

The purpose of this study was examine the relationship between physical motions characteristics and the subjective impressions produced in dancing.
The structure of subjective impressions for dance motions with no regular rule of operation was investigated using Okinawan folk dance "Kachaasii". Here was noticed only the movement of the leg . First, the questionnaire method and the principal component analysis were adopted for investigation of the mental structure of the leg motion in dance. Next, the dance was reproduced using CG, and were examine subjective impression. The results, the impression given only in the movement of the leg motion was extracted.

Key Words: motion characteristics, sensitivity element, free-style dance, motion of legs.

1. introduction

Humans have various impression for the movement that they observed. For example, be "beautiful behavior" or the "brisk movements". The study about subjective impressions structure of observer has been done in order to evaluate human motions and image for design of a thing quantitative. In the field of the dance, research has been done in order to survey impression structure for a dance. However, there is little as for the research which showed the quantitative analysis of relationship between impression and motion characteristics in from a point of view of dance motions. Furthermore, be not suggested on motion characteristic to give one specified impression to an observer either. The human impressions and physical characteristics relationship have been investigated for the music and a picture. Nevertheless, there has been no study that tried to make relationship clear quantitatively about motion characteristics of humans and subjective impressions. As for it, two cause is considered. The point that it is difficult for quantitative analysis to do movement of humans such as sign language or a gesture represented by non-verbal communication as motion characteristic. Furthermore, as for determining impression of human, there are a facial expression and an opinion that other factor is strong. Another one is the point that it is difficult to show subjective impression of humans and relationship with a physical characteristic quantitatively.

The purpose of this study is to examine the relationship between physical motion characteristics and the subjective impressions produced in dancing. The structure of subjective impressions for dance motions with no regular rule of operation was investigated using Okinawan folk dance "Kachaasii".

In our precedence research, the relationship between impression and kinetic characteristics which noticed the arm motion in the dance has been examined. The results showed that Kachassii has five subjective scales: preference, hard - soft, dynamic - static, monotonous - peculiar and balanced - large. Furthermore, there were highly correlation between those scales and motion features respectively as follows: preference vs. turnaround of wrist, hard vs. pause of movement, soft vs. no pause of movement, dynamic vs. height of movement, static vs. phase movement from forearm to wrist, monotonous vs. horizontal movement of

wrist, peculiar vs. horizontal angular velocity of shoulder. accented vs. vertical movement of wrist, and large vs. flexion and extension movement of elbow. In this way that "movement of arm and a fingertip" are connected with the upper limbs motion is shown in some impression [1]-[3]. However, what the impression that legs motion movement except the upper limbs affects exists in is suggested. Only the movement of the leg was discussed here. First, the questionnaire method and the principal component analysis were adopted for investigation of the mental structure of the leg motion in dance. Next, the dance was reproduced with Computer Graphics, and the subjective impressions were examined. The results showed that the movement of leg has five subjective scales: preference, weight, width and kinetics.

2. Subjective impression experiment

2.1. Methods

The purpose of the impression evaluation experiment here in this chapter is to know the impressions to various patterns of dance motion and to examine the impression structure of free-style dance Kachaasii with principal component analysis. The video image was prepared for the evaluation in order to carry out the impression evaluation of Kachaasii. At first a lot of dance patterns were collected without relationship in presence of dance experience, age and sex. Dance motions of 23 patterns were selected by motion characteristics pattern. The various dance patterns performed by the same dancer were recorded at the front with video camera. The size and width of dancer's body was uniformed in each video sequence. The face was hidden with dark-color veil so as to avoid the impression deviated by dancer's sexuality, face or facial expression. The questionnaire was carried out with thirty pairs of adjectives and the antonyms. The adjectives are shown in table.1 and fig.1. Twenty-three patterns of dance were prepared for subjective evaluation as video image. The example is shown in fig.2. Sufficient trial time was allowed for subjects to watch and answer in each evaluation of dance. The analytical results by principal component analysis were as follows: eigenvalue, proportion and cumulative proportion of each component are shown in table2. The first through five components seemed to be empirically meaningful, mainly from the view point of proportion. Then, the values of each factor loading were examined in first through five components. Impression characteristics of each component are shown in fig.3. Each component have impressions characteristics as follows: the first component exhibited "preference", the second component "heavy - light", the third component "peculiar - monotonous", the fourth component "dynamic - static ", and the fifth "large - balanced".

Table 1 Adjectives for principal

(1)	speedy	(16)	accented
(2)	regular	(17)	straight
(3)	stabilized	(18)	rising
(4)	strong	(19)	heavy
(5)	dynamic	(20)	cheerful
(6)	beautiful	(21)	skillful
(7)	graceful	(22)	harmonious
(8)	like	(23)	wide
(9)	magnificent	(24)	good
(10)	profound	(25)	rhytmical
(11)	smooth	(26)	complex
(12)	large	(27)	balanced
(13)	toward	(28)	delight
(14)	hard	(29)	thick
(15)	forceful	(30)	peculiar

Fig.2 An example of video image

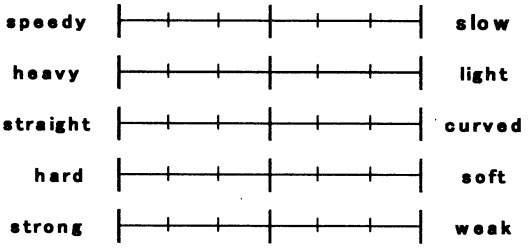


Fig.1 An example questionnaire

Table2. Eigenvalues and propotions of principal component analysis

principal component	eigenvalue	propotion	comulative propotion
1st	7.42	24.73%	24.73%
2nd	4.82	16.06%	40.79%
3rd	3.11	10.36%	51.15%
4th	2.12	7.06%	58.21%
5th	1.40	4.67%	62.88%
6th	1.15	3.83%	66.70%

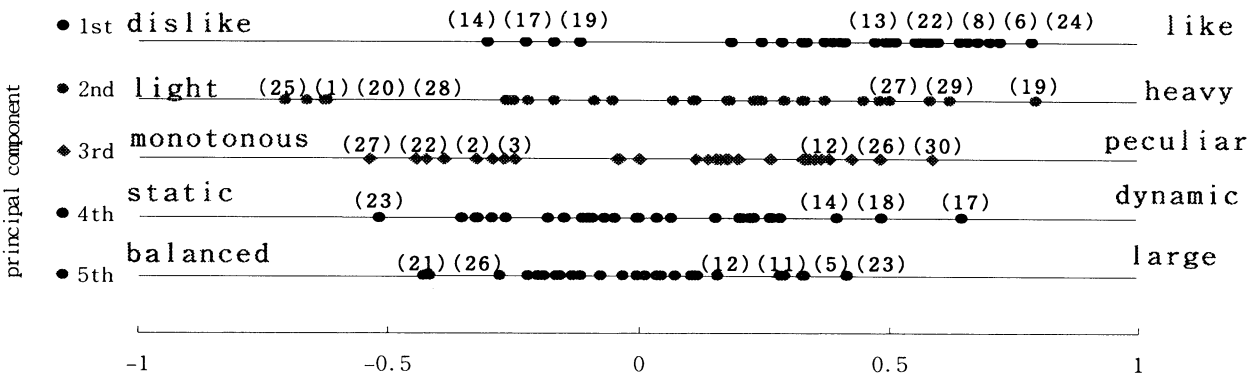


Fig.3. Adjectives selected in each component.

2.2. Result of the multiple regression analysis

A statistical measure as a principal component score is also obtained by the principal component analysis, which is an averaged score in each dance pattern weighted by each value of factor loadings obtained at word pairs in each principal component, through which the degree of a certain kind of impression of a dance pattern can be clarified. The larger a principal component score is, the greater impact a dance pattern has in a certain kind of impression represented by each principal component. In this chapter, therefore, kinematics features were parameterized in each dance pattern, and the relationship between the motion features as explanatory variables and subjective impressions as response variables were quantified by the multiple regression analysis. As a result of analysis, the relationship between impression and kinetic characteristics which noticed the leg motion in the dance has been examined. The result of multiple regression analysis is shown by table3. There were highly correlation between those scales and motion features respectively as follows: preference vs. flexion and extension movement of knee and hip joint, heavy vs. point of hip joint, peculiar vs. movement of the tip of a foot, monotonous vs. no flexion and extension of knee, dynamic vs. movement of fling up legs, large vs. horizontal movement of legs, and accented vs. recomposition movement of legs.

Table3. Result of the multiple regression analysis

first principal component: like—dislike

motion characteristics	Comulative propotion	Regression coefficient	F Value	p
Knee angular variance	36.18%	0.76	11.90	**
Hip jpoint angular skewness	49.00%	-0.33	5.03	*
Hip jpoint angular average	59.43%	0.36	4.88	*

second principal component: heavy—light

motion characteristics	Comulative propotion	Regression coefficient	F Value	p
Ankle angular average	13.17%	0.64	3.18	
Knee angular average	23.77%	0.66	2.78	
Knee angular variance	43.72%	0.70	6.73	*
Hip jpoint angular skewness	51.44%	0.34	2.86	

third principal component: peculiar—monotonous

motion characteristics	Comulative propotion	Regression coefficient	F Value	p
Ankle angular variance	29.92%	0.25	8.97	**
Hip jpoint angular skewness	41.60%	-0.44	4.00	
Hip jpoint angular average	57.01%	-0.42	6.81	*

fourth principal component: dynamic—static

motion characteristics	Comulative propotion	Regression coefficient	F Value	p
Hip jpoint angular variance	50.10%	0.64	21.09	**
Ankle angular skewness	60.48%	0.45	5.25	*
Ankle angular variance	65.92%	-0.43	3.04	
Hip jpoint angular skewness	69.51%	-0.32	2.11	
Hip jpoint angular average	74.77%	-0.30	3.55	

fifth principal component: large—balanced

motion characteristics	Comulative propotion	Regression coefficient	F Value	p
Hip jpoint angular average	28.43%	-0.71	8.34	**
Ankle angular skewness	59.80%	0.47	15.61	**
Knee angular variance	64.43%	0.08	2.47	

* $p < 0.05$ ** $p < 0.01$

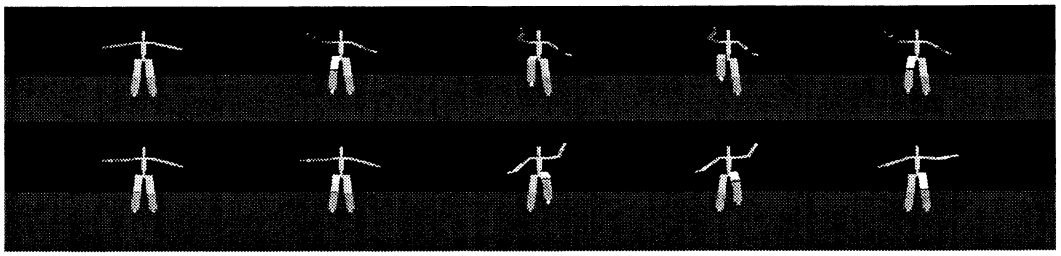


Fig.4 An example of CG animation dance image

3. Subjective impression experiment by CG

The motion characteristics was reproduced using CG. And there were examined subjective impression. The example is shown by CG animation in fig.4. The example by CG animation is shown in fig.4. The analytical results by principal component analysis using CG image were as follows: eigenvalue, proportion and cumulative proportion of each component are shown in Table 4. Then, the values of each factor loading were examined in first through five components. Impression characteristics of each component are shown in Fig.5. Each component have impressions characteristics as follows: the first component exhibited "preference", the second "heavy - light ", the third "balanced - large ", the fourth "dynamic- static ", and the fifth "hard - soft ".

Table4. Eigenvalues and propotions of principal component analysis by CG

principal component	eigenvalue	propotion	comulative propotion
1st	8.23	27.42%	27.42%
2nd	4.95	16.50%	43.92%
3rd	2.41	8.02%	51.94%
4th	2.16	7.20%	59.14%
5th	1.31	4.37%	63.51%
6th	1.14	3.80%	67.31%

4. Results and discussion

In this experiment, extracted motion characteristics of legs related to sensitivity of human. Impression of legs motion characteristics by human, "like - dislike", "heavy - light", "peculiar - monotonous", "dynamic - static", and "balanced - large" was extracted. And so, Impression of legs motion image by CG , "like - dislike", "heavy - light", "balanced - large ", "dynamic- static ", and "hard - soft " was extracted. As a result it was suggested that "like - dislike", "heavy - light", "balanced - large", "dynamic - static" were impression provided only by movement of legs.

5. References

[1] S.Kamisato and K.Hoshino, Motion characteristics of finger and wrist trajectories and subjective impression in a free-style dance: IEICE HIP, vol.100, no.138, pp.47--51, 2000.

[2] S.Kamisato and K.Hoshino, Evaluation of subjective impressions and motion characteristics in a free-style dance: Proc. of ORSJ, vol.2, pp.309--314, 2000.

[3] S.Kamisato, S.Odo, K.Yamada and K.Hoshino, Quantitative analysis of relation between subjective impressions and characteristics of arm motion in free-style dance. ANNIE, Nov., 2003.

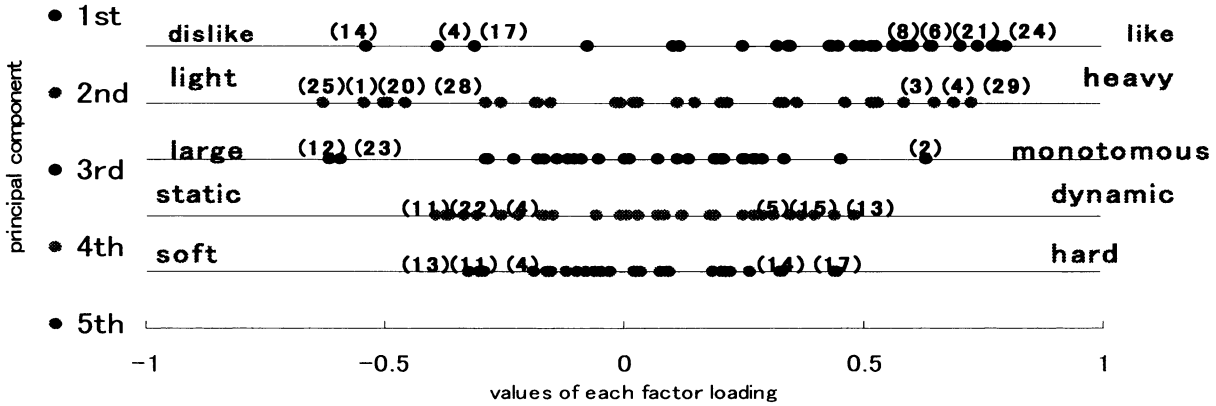


Fig.5. Adjectives selected in each component by CG animation.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Development of Vietnamese pronunciation training system by DP matching method

MINH Nguyen Tan
Dept. of Computer Science
Tokyo Institute of Technology
2-12-1 Ookayama, Meguro
Tokyo, 152-8550, JAPAN

JUN Murakami
Dept. of Information and Computer Sciences
Kumamoto National College of Technology
2659-2 Suya, Nishigoshi
Kumamoto, 861-1102, JAPAN

Abstract

This paper describes the development of a computer assisted language learning system for learning of Vietnamese pronunciation using DP matching method. When a learner pronounces a Vietnamese word toward a microphone, the system computes the correctness of his pronunciation by comparing with the standard one, and displays that correctness by points at once. The learner can master the correct pronunciation by repeating a practice. From the results of the experiments, we confirmed that the learner could be trained to improve his pronunciation by using our system. The system is described by the Visual C++ language to provide a GUI (Graphical Use Interface) environment for the users.

1 Introduction

Vietnamese is classified as a monosyllabic and tonal language, which means that all words are only one syllable long. Each syllable of the words can be pronounced in six different tones and those tones confer different meanings on the words. It is quite difficult for non-native speakers to pronounce them with clear distinction.

Since these tones are particularly difficult for Japanese to pronounce, we developed the Vietnamese pronunciation training system. Our system gives the standard pronunciation of Vietnamese words by native Vietnamese speaker to the learner and compares his pronunciation to the standard one by using the DP (Dynamic Programming) pattern matching technique [1]. When a learner pronounce a Vietnamese word toward a microphone after hearing the standard pronunciation, the system computes the correctness of his pronunciation by comparing with the standard one, and displays that correctness by points.

From the results of the experiments, we confirmed that the learner could be trained to improve his pronunciation of the Vietnamese words by using the system.

2 Design of the system

We designed the system as follows.

(1)The system is constituted of some common elements, such as a personal computer, a microphone and an earphone to provide the system for anyone who wish to use.

(2)The system provides a GUI (Graphical User Interface) environment for users.

To satisfy those conditions, we used Visual C++ programming language to describe the system. We confirmed that the system would work normally under following environment:

OS : Windows 98SE/2000/XP.
RAM : more than 64 MB.
CPU : more than 500 MHz.

2.1 Structure of the system

Figure 1 shows the constitution of the system. The system consists of the input module and the recognition module. The input module acquires the learner's pronunciation of a Vietnamese word, and then analyzes the word to generate the standard pronunciation. The recognition module analyzes both the acquired pronunciation and the standard one to compute the degree of a similarity between those two pronunciations by using a pattern matching technique.

2.2 The flow of the system

Figure 2 shows the user interface window of the system. When a user inputs a Vietnamese word to the system from the keyboard, he can hear the standard pronunciation of the word through the speaker. Then, he pronounces that word toward the system. The system computes the correctness of user's pronunciation

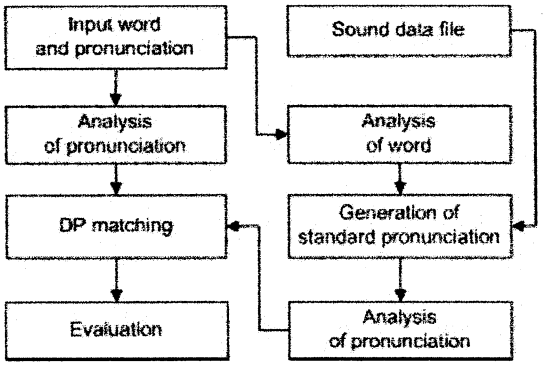


Figure 1: System construction.

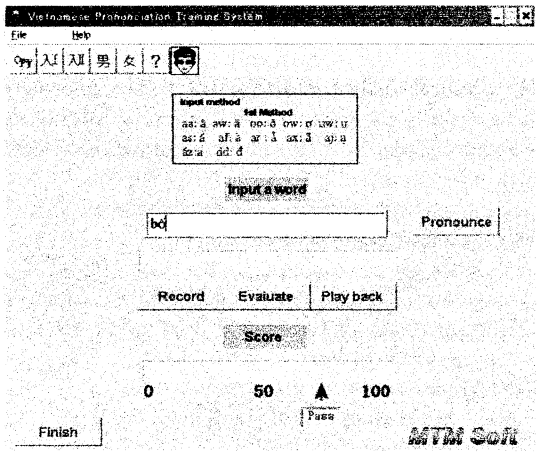


Figure 2: System interface window.

by comparing with the standard one, and displays that correctness by points at once. The user can master the correct pronunciation of the Vietnamese word by repeating this flow.

3 Details of the system

3.1 Speech data file

The construction of speech data file is shown in Figure 3. The sound data of the standard pronunciation of some Vietnamese words is stored in the speech data file. The sound data is stored after being decomposed into vowel sounds, consonant sounds and tunes of voice because actual data size of the pronunciation is very large. It is needed to store some hundreds of phonemes in the data file to generate the pronunciation of almost all of the Vietnamese words. As the data size of a phoneme is about 10 to 15 kB when the sound data is sampled at 11025 Hz in the 16 bit monaural Wave form, the speech data file requires approximately 10 MB.

Practically, we used speech data files, which are named sound.dat and wordpos.dat. In the former file, the sound data of all of the phonemes are stored. The size and the position of each phoneme in the wordsound.dat are stored in the latter file. The size and position of a phoneme are taken out from wordpos.dat at the beginning. And then the sound data can be obtained from wordsound.dat by using acquired information about the phoneme.

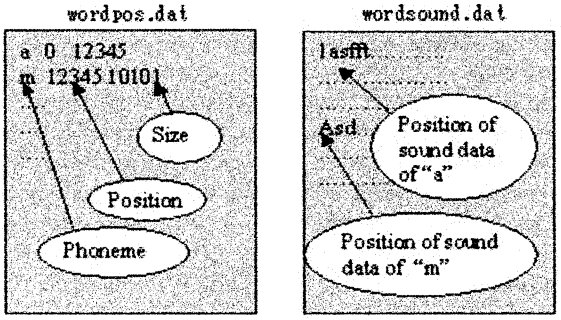


Figure 3: Speech data file.

3.2 Generation of standard pronunciation

Figure 4 shows the flow of a generation of the standard pronunciation for the input Vietnamese word. The user can hear the obtained standard pronunciation repeatedly in order to check his own pronunciation with the standard one.

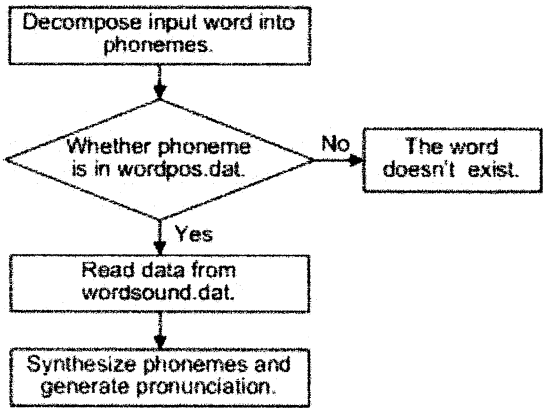


Figure 4: Overview of standard pronunciation generation process.

3.2.1 Decomposition of input word into phonemes

The following are characteristics of Vietnamese language.

(1) Every word can be decomposed into one or two phonemes. The words which begin with a consonant are divided into two phonemes, and the other words become a phoneme themselves. There are no words that have more than three phonemes.

(2) There are 26 combinations of consonants to appear in the beginning of a word.

By exploiting above characters, the system decomposes the input words into phonemes.

3.2.2 Synthesis of standard pronunciation

After the process of preceding section, the system reads the size and the position of the phonemes in wordsound.dat from wordpos.dat. And then the system can obtain the sound data of those phonemes from wordsound.dat.

The sound data can be used directly as a standard pronunciation as for one-phoneme-words. For other case, two corresponding sound data of phonemes are combined to make standard pronunciation. In the latter case, obtained standard pronunciations might sound unnaturally because of a difference of pitches of the two phonemes. In order to solve this problem, the pitches of those phonemes are adjusted to the pitch of the second phoneme, because the pitch of a two-phonemes-word depends on that of the second phoneme. To adjust those pitches, the fast algorithm in the reference [2] is adopted. Figure 5 shows the flow of above procedure, and Figure 6 demonstrates an example of which the pronunciation of the Vietnamese word [ma] is synthesized from the phonemes [m] and [a].

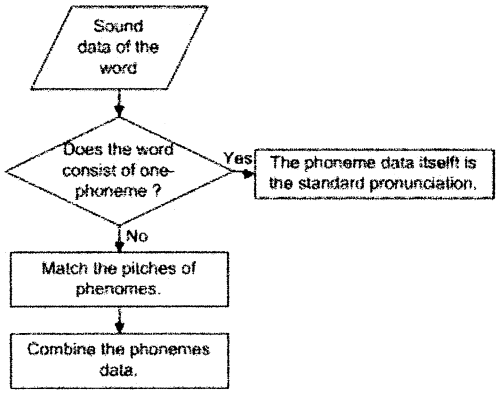


Figure 5: Segmentation and synthesis of pronunciation data.

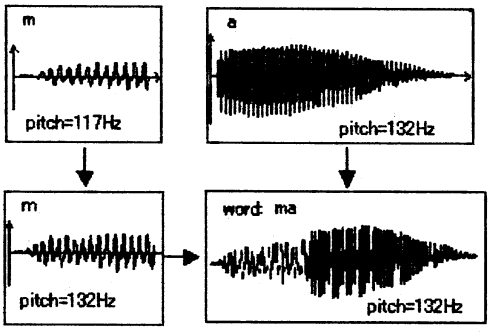


Figure 6: Example of synthesized pronunciation data.

3.3 Recognition module

The degree how the learner's pronunciation of the word corrects in comparison with the standard pronunciation of that word is computed in the recognition module. To evaluate that degree, the cepstrum analysis [3] and the DP matching method are exploited.

3.3.1 Voice segment detection and the cepstrum analysis

To begin with, the section where the voice signal of learner's pronunciation is exist has to be detected. Then we tried to detect the voice segment exactly by using the energy and the number of zero crosses [4] of the voice waveform to compare with the standard pronunciation, though it is difficult to detect this segment.

The cepstrum is obtained as a result of the inverse Fourier transformation of a logarithm of the power spectrum of a voice waveform. The pitch and the envelope of the spectrum of a voice waveform are calculated by using this cepstrum. The obtained cepstrums is treated as vector numbers and stored in arrays.

3.3.2 Pattern recognition by DP matching method

The DP matching method is usually adopted in order to regularize some time series of cepstrums relating the time axis nonlinearly. The distance between the cepstrums of the learner's pronunciation and the standard pronunciation is calculated by using this method.

Those two time series of the cepstrum that are obtained in the previous section are denoted respectively as

$$\boldsymbol{A}= \boldsymbol{a}_1,\boldsymbol{a}_2,\boldsymbol{a}_3,\ldots\boldsymbol{a}_i\ldots,\boldsymbol{a}_N,$$
$$\boldsymbol{B}= \boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_3,\ldots\boldsymbol{b}_i\ldots,\boldsymbol{b}_M,$$

where the vector $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ is the cepstrum of a frame.

The distance between the cepstrum $\boldsymbol{a_i}$ and $\boldsymbol{b_j}$ is given by

$$d(i,j) = \sum_k d(a_i[k], b_j[k]),$$
$$d(a_i[k], b_j[k]) = \|a_i[k] - b_j[k]\|^2, \quad (1)$$

where $a_i[k]$ and $b_j[k]$ are the k-th elements of each vectors. Here we introduce $R(k)$ which is defined as

$$R(k) = \min \begin{pmatrix} R(k-1) + d(i-1,j) + d(i,j) \\ R(k) + \frac{4}{3}\{d(i,j) + d(i-1,j-1) \\ \quad +d(i-2,j-2)\} \\ R(k+1) + d(i,j-1) + d(i,j) \end{pmatrix},$$
$$i + j = 3l + 2, l = 0, 1, ..l_{max}, l_{max} = [\frac{N+M-2}{3}], \quad (2)$$

where $l = 3$, and initialized value of $R(k)$ is 0. From the equation (1) and (2), the distance between A and B is defined as

$$D(\boldsymbol{A}, \boldsymbol{B}) = \frac{\min R(k)}{N+M}, \quad k = i - j, i + j = N + M. \quad (3)$$

By using the distance $D(\boldsymbol{A},\boldsymbol{B})$, the pronunciation of a learner can be evaluated. The smaller this distance becomes, the better the learner improve his pronunciation

4 Experimental results

Experiments are performed to evaluate the system on the conditions below.

(1)Make sure that the learner is in a quiet room to avoid an influence of a noise.

(2)The system consists of a personal computer (CPU : Celeron 800 MHz, RAM : 128 MB, OS : Windows 2000), a microphone and speakers.

(3)The learners practice Vietnamese pronunciations repeatedly according to the way which is described in section 2.2.

The experiments are carried out to three subjects. Each subject pronounces a one-phoneme-word and two-phonemes-word (shown in Table 2) five times respectively. Table 1 shows the scores of the experiments. From the table 1, we can find that the score doesn't increase monotonously, because of both the influence of the subjects' breathing sound mixed in the pronunciations and the insufficient accuracy of the recognition process. It can be seen also that the fifth score is higher than the first one in all the experiments. This means that subjects had made progressed considerably in Vietnamese pronunciation. On the other hand, we can see that the subject could not advance his score easily for the vowels which the Japanese language doesn't have.

Table 1: Experimental result.

Subject	A		B		C	
Word	o	na	a	ma	ố	bố
1st time	50	30	50	80	60	55
2nd time	70	70	40	100	70	60
3rd time	100	60	60	90	90	80
4th time	90	100	70	100	80	75
5th time	100	85	100	100	95	80
Average score	82	69	64	94	79	70

Table 2: Vietnamese words which is used for the experiments and their meanings.

One-phoneme word	Two-phonemes word
a : letter "a"	na : custard-apple
o : aunt	ma : ghost
ố : stain	bố : father

5 Conclusions

In this paper, we described a design, an implementation and an evaluation of our Vietnamese pronunciation training system. From the results of the experiments, we showed that the subjects' pronunciations approach to the standard pronunciations gradually by repeating a practice, while there are some rooms for improvement on the system about the influence of a noise and the accuracy of the recognition process.

References

[1] V. Vintsyuk, "Speech Recognition by Dynamic Programming," Kybernetica, Vol. 4, No. 1, pp.81-88, Jan. 1968.

[2] Hiroshi Toda, "Sound Effect", *CMagazine* (in Japanese),Vol.8, No. 12, pp. 22-55, Dec. 1996.

[3] A. M. Noll, "Short-Time Spectrum and Cepstrum Techniques for Vocal-Pitch Detection" , J. Acoust. Soc. Amer., Vol. 36, No. 2, pp.296-302, 1964.

[4] L. R.Rabiner, R. W.Schafer. *Digital Processing of Speech Signals*, Prentice-Hall, 1978.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Fault Diagnostic of Induction Motors for Equipment Reliability and Health Maintenance Based upon Fourier and Wavelet Analysis

Hyeon Bae, Youn-Tae Kim, Sang-Hyuk Lee, Sungshin Kim, and Man Hyung Lee*

School of Electrical and Computer Engineering, Pusan National University, Busan, Korea
*School of Mechanical Engineering, Pusan National University, Busan, Korea

Pusan National University 30 Changjeon-dong, Keumjeong-ku, Busan 609-735, Korea
E-mail: baehyeon@pusan.ac.kr, leehyuk@pusan.ac.kr, sskim@pusan.ac.kr, mahlee@pusan.ac.kr

Abstract – A motor is the workhorse of our industry. The issues of preventive and condition-based maintenance, online monitoring, system fault detection, diagnosis, and prognosis are of increasing importance. Different internal motor faults along with external motor faults are expected to happen sooner or later. This paper introduces the fault detection technique of induction motors based upon the stator current. The fault motors have rotor bar broken or rotor unbalance defect, respectively. The stator currents are measured by the current meters and stored by the time domain. The time domain is not suitable to represent the current signals, so the frequency domain is applied to display the signals. The Fourier Transformer is used for the conversion of the signal. After the conversion of the signals, the features of the signals have to be extracted by the signal processing methods like a wavelet analysis, a spectrum analysis, etc. The discovered features are entered to the pattern classification model such as a polynomial neural network model. This paper describes the fault detection results that use wavelet decomposition. The wavelet analysis is very useful method for the time and frequency domain each. Also it is powerful method to detect the feature in the signals.

Keywords: fault detection, Feature extraction

1. Introduction

The most popular way of converting electrical energy to mechanical energy is an induction motor. This motor plays an important role in modern industrial plants. The risk of motor failing can be remarkably reduced if normal service conditions can be arranged in advance. In other words, one may avoid very costly expensive downtime of plant by proper time scheduling of motor replacement or repair if warning of impeding failure can be obtained in advance. In recent years, fault diagnosis has become a challenging topic for many electric machine researchers.

The major faults of electrical machines can broadly be classified as follows [1]:

- Abnormal connection of the stator windings
- Broken rotor bar or cracked rotor end-rings
- Static and/or dynamic air-gap irregularities
- Bent shaft (akin to dynamic eccentricity)
- Shorted rotor field winding
- Bearing and gearbox failure

Faults in electric machines produce one of more of the following symptoms:

- Unbalanced air-gap voltages and line currents
- Increase torque pulsations
- Decreased average torque
- Increase losses and reduction in efficiency
- Excessive heating

The diagnostic methods to identify the above faults may involve several different types of fields of science and technology [1], [2]. Several methods are applied to detect the faults in induction motors as the following:

- Electromagnetic field monitoring
- Temperature measurements
- Infrared recognition
- Radio frequency (RF) emissions monitoring
- Noise and vibration monitoring
- Chemical analysis
- Acoustic noise measurements
- Motor current signature analysis (MCSA)
- Model, AI and NN based techniques

Although the Fourier transform is an effective method and widely used in signal processing and the transformed signal may lose some time domain information. The limitation of the Fourier transform in analyzing non-stationary signals lead to the introduction of time-frequency or time scale signal processing tools, assuming the independence of each frequency channel when the original signal is decomposed. This assumption may be considered as the limitation of this approach.

Wavelet transform is a method for time varying or non-stationary signal analysis, and uses a new description of spectral decomposition via the scaling concept. Wavelet theory provides a unified framework for a number of techniques, which have been developed for various signals processing applications. One of its feature is multi-resolution signal analysis with a vigorous function of both time and frequency localization. This method is effective for stationary signal processing as well as non-stationary signal processing. Mallat's pyramidal algorithm based on

convolutions with quadratic mirror filters is a fast method similar to FFT for signal decomposition of the original signal in an orthonormal wavelet basis or as a decomposition of the signal in a set of independent frequency bands. The independence is due to the orthogonality of the wavelet function [3].

2. Fault detection of Induction Motor

2.1 Bearing Faults

Though almost 4~50% of all motor failures are bearing related, very little has been reported in the literature regarding bearing related fault detection techniques. Bearing faults might manifest themselves as rotor asymmetry faults from the category of eccentricity related faults [4]. The vibration frequency of the fault is as follows,

$$f_1[Hz] = (f_r/2) f_r [1 - b_d \cos(\beta)/d_p] \quad (1)$$

where f_r is the rotational frequency, N is the number of balls, b_d and d_p are the ball diameter and ball pitch diameter respectively, and β is the contact angle of the ball. The following equation includes the vibration frequency and current spectrum.

$$f_{bng} = | f_1 \pm m.f_v | \quad (2)$$

where m=1, 2, 3, ... and f_v is one of the characteristic vibration frequency.

Artificial intelligence or neural networks have been researched to detect bearing related faults on line. And also adaptive, statistical time frequency method are studying to find bearing faults.

2.2 Broken rotor bar and end ring faults

Rotor failures now account for 5-10% of total induction motor failures. Broken rotor bars give rise to a sequence of side bands given by:

$$f_b = (1 \pm 2ks) f \qquad k = 1, 2, 3, \ldots \quad (3)$$

where f is the supply frequency and s is the slip. Frequency domain analysis and parameter estimation techniques have been widely used to detect this type of faults.

In practice, the current side bands around fundamental may exist even when the machine is healthy [5]. Also rotor asymmetry, resulting from rotor ellipticity, misalignment of the shaft with the cage, magnetic anisotropy, etc. shows up at the same frequency components as the broken bars [6]. Therefore other features of this fault need to be investigated.

2.3 Eccentricity related faults

This fault is the condition of unequal air-gap between the stator and rotor. It is called static air-gap eccentricity when the position of the minimal radial air-gap length is fixed in the space. This maybe caused by the ovality of the stator core or by the incorrect positioning of the rotor or stator at the commissioning stage.

In case of dynamic eccentricity, the center of rotor is not at the center of rotation, so the position o minimum air-gap rotates with the rotor. This maybe caused by a bent rotor shaft, bearing wear or misalignment, mechanical resonance at critical speed, etc. In practice an air-gap eccentricity of up to 10% is permissible. Both static and dynamic eccentricities tend to exist in practice.

Using MCSA the equation describing the frequency components of interest is:

$$f\left[(kR \pm n_d)\frac{(1-s)}{p} \pm v\right] \quad (4)$$

where n_d=0 in case of static eccentricity, and n_d=1, 2, 3, ... in case if dynamic eccentricity f is fundamental supply frequency, R is the number of rotor slots, s is slip, p is the number of pole pairs, k is any integer and υ is the order of the stator time harmonics.

Other equations are also presented in the literature as low frequency components for mixed eccentricity [5]. As it is obvious, sometimes, different faults produce nearly the same frequency components or behave like healthy machine, which make the diagnosis impossible. This is the reason why new techniques must also be considered to reach a unique policy for distinguishing among faults. Park's vector based upon voltage and current has been proposed to detect the motor fault.

3. Wavelet Transformation

A wavelet is a function ψ belonging to $L^2(R)$ with a zero average. It is normalized and centered in the in the neighborhood of t=0. A family of time-frequency atoms is obtained by scaling ψ by a^j and translating is by b:

$$\psi_{a,b} = |a|^{\frac{-j}{2}} \psi\left(\frac{t-b}{a^j}\right) \quad (5)$$

These atoms also remain normalized. The wavelet transform of f belonging to $L^2(R)$ at the time b and scale a^j is:

$$Wf(b,a^j) = \langle f, \varphi_{b,a^j} \rangle = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{a^j}} \varphi^*\left(\frac{t-b}{a^j}\right) dt \quad (6)$$

A real wavelet transform is complete and maintains an energy conservation as long as the wavelet satisfies a weak admissibility condition which is:

$$\int_0^{+\infty} \frac{|\Psi(w)|^2}{|w|} dw = \int_{-\infty}^{0} \frac{|\Psi(w)|^2}{|w|} dw = C_\psi < +\infty \quad (7)$$

When $Wf(b, a^j)$ is known only for $a<a_0$ to recover f we need a complement of information corresponding to $Wf(b, a^j)$ for $a<a_0$. This is obtained by introducing a *scaling function* ϕ that is an aggregation of wavelet at scales larger than 1. $\hat{\psi}(w)$ and $\hat{\varphi}(w)$ are Fourier transforms of $\psi(t)$ and $\varphi(t)$ respectively. $\psi(t)$ is a band pass filter, and $\varphi(t)$ is a low-pass filter. Taking positive frequency into account $\hat{\varphi}(w)$ has information in $[0, \pi]$ and $\hat{\psi}(w)$ in $[\pi, 2\pi]$. Therefore they both have complete signal information without any redundancy.

Decomposition of the signal in [0, π] using Mallat's algorithm gives:

$$h(n) = \langle 2^{-j}\varphi(2^{-1}t)\varphi(t-n)\rangle$$
$$g(n) = \langle 2^{-j}\psi(2^{-1}t)\varphi(t-n)\rangle \qquad j = 0,1,2,\ldots \quad (8)$$

Wavelet decomposition does not involve the signal in [π, 2π]. In order to decompose the signal in whole frequency band, wavelet packet can be used for this purpose. After decomposition for l times, we will get 2^l frequency bands each with the same bandwidth. That is:

$$\left[\frac{(i-1)f_n}{2}, \frac{if_n}{2}\right] \qquad i = 1,2,\ldots,2^j \quad (9)$$

where f_n is the Nyquist Frequency, in the i^{th} frequency band. Wavelet packet de-composes the signal into one low-pass filter $h(n)$ and 2^l-1 band-pass filters $g(n)$, provides diagnosis information in 2^l frequency bands.

Functions $h(n)$ and $g(n)$ can be obtained by inner product of $\psi(t)$ and $\varphi(t)$.

$$h(n) = \langle 2^{-j}\varphi(2^{-1}t)\varphi(t-n)\rangle$$
$$g(n) = \langle 2^{-j}\psi(2^{-1}t)\varphi(t-n)\rangle \qquad t \in R, n \in Z \quad (10)$$

$$A_j(n) = \sum_k h(k-2n)A_{j-1}$$
$$D_j(n) = \sum_g g(k-2n)A_{j-1} \qquad n = 1,2,3,\ldots \quad (11)$$

where $A_0(k)$ is the original signal and A_j is the low frequency approximation at the resolution j. D_j is called high frequency detail signal. After de-composition of j time, we can obtain one approximation signal A_j and D_1, D_2, ..., D_j detail signals.

Wavelet packet decomposition is:

$$x_{2n}(t) - \sqrt{2}\sum_k h(k)x_n(2t-k)$$
$$x_{2n+1}(t) - \sqrt{2}\sum_k g(k)x_n(2t-k) \quad (12)$$

where $x_1(t)$ is the original signal. Comparing (12) with (10), we can find that A_j in (10) is decomposed but also D_j in (10) is decomposed in (12).

Wavelet and wavelet packet decompose the original signal that is non-stationary or stationary into independent frequency bands with multi-resolution. Now we have an effective tool to monitor broken bar fault in the rotor.

4. Experimental Results

4.1 Current Signals and Data Preprocessing

Motor rating applied in this paper is dependent on the electricity conditions. The rated voltage, speed, and horsepower are 220V, 3450RPM, and 0.5HP, respectively. And implemented motor specification includes the number of slot, the number of pole, slip, etc. The specification of used motor is 34 numbers of slots, 4 numbers of poles, and 24 numbers of rotor bars. The slip is determined by calculating an actual motor speed and a rated speed. The specification of measured input current signals under this condition consists of 16,384 sampling numbers, 3kHz maximum frequency, and 2.1333 measuring time. Applied fault types in this study are broken rotor, faulted bearing, bowed rotor, unbalance, and static and dynamic eccentricity case.

If the wavelet decomposition is implemented in the fault detection of induction motors, the unsynchronized current phase problem should influent the detection results much. If target signals are not synchronized, the unexpected results will appear in the wavelet decomposition. Therefore the signals are re-sampled by synchronizing signals with phase 0. And the average value divided by one cycle signal is calculated to reduce the noise of original signals as shown in Fig. 2.

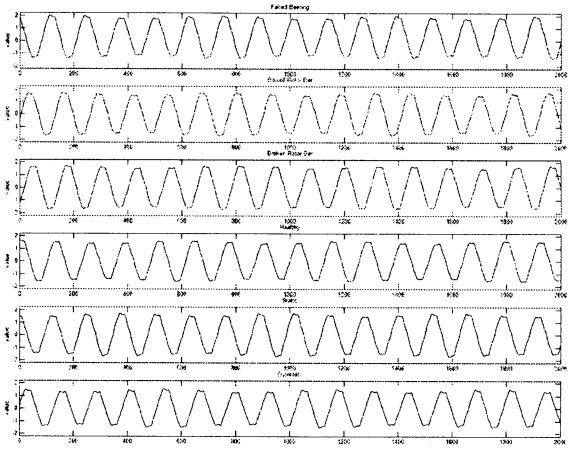


Fig. 1. Original current signals.

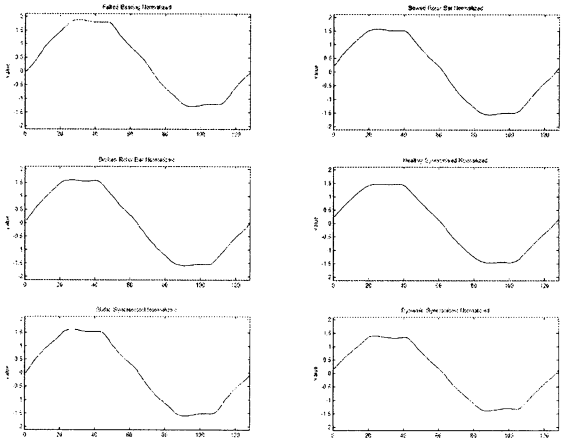


Fig. 2. Data synchronizing and filtering.

4.2. Feature Extraction

In this paper, two methods are considered to detection the faults of the induction motor. The Fourier and wavelet transform are applied in full load conditions. In this conversion, some features to faults can appear but in our test, the case of broken rotor bar has a different feature that is a specific side-bend around supply frequency shown in Fig. 3. But the other faults could not

be classified by the FFT. Thus the other approach is required to detect the faults. Wavelet transform is implemented for the troubleshooting. In this paper, the specific scale of wavelet decomposition is considered to improve the performance. Because this detail scale contains most information of several faults, 6^{th} decomposition scale of 12 scales is analyzed for the fault detection. And Coif-let wavelet function that is determined by the experimental experience for the better performance is used in the decomposition.

The features are extracted from the 6th decomposition. The collected features are a gradient value between the 3^{rd} and 4^{th} values and a peak value of 4^{th} of the 6th decomposition result. Fig. 6 represents the results of wavelet analysis. From this result, it is possible that the wavelet decomposition is suitable for the fault detection in the induction motors.

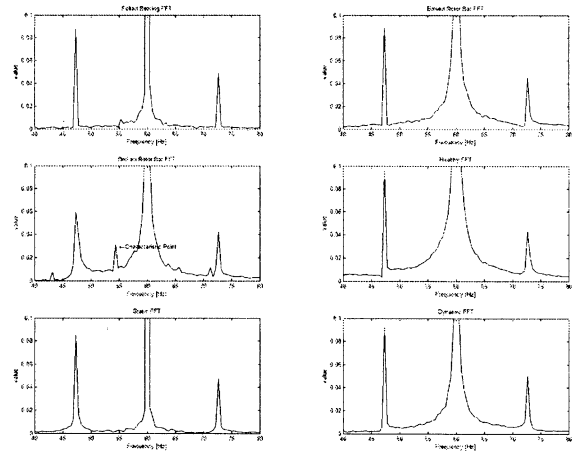


Fig. 3. FFT result of broken rotor bar and healthy case.

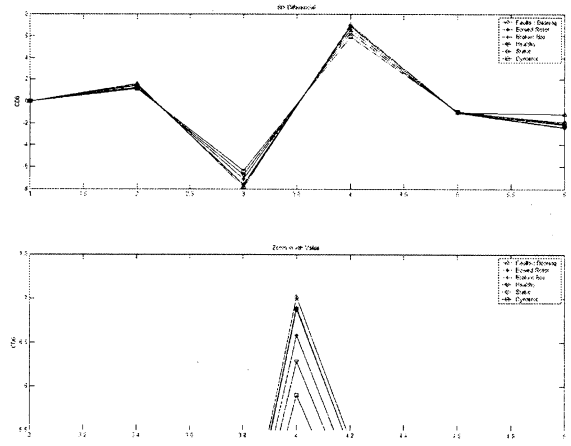


Fig. 4. Wavelet analysis in healthy and faulty motor.

4.2. Fault Classification

Input variables consist of 6 numbers shown in Table 1. Table 2 shows the brief results of classification by using dynamic polynomial neural networks. From the results, we can examine that the extracted variables are suitable for the fault detection of the induction motors.

Table 1. Input variable for the classification.

	Inputs		
Fourier transform	Left side band	Center side band	Right side band
Wavelet transform	Left gradient	Peak value	Right gradient

Table 2. Results of the fault detection.

Fault kinds	Target values	Test results
Bearing fault	1	1.002311924
		1.002311925
		1.002311924
Broken rotor bar	3	2.949762394
		2.949762394
		2.949762395
Static eccentricity	5	4.820981697
		4.820981699
		4.820981699

5. Conclusion

The wavelet analysis is possible method to detect faults of induction motors except one mechanical fault like bearing bowed fault. The gradient and peak values of the 6^{th} detail result of 12 scales of wavelet decomposition are applied.

In the wavelet analysis, features of broken rotor bar and static eccentricity are similar but the result of Fourier is much different. Therefore applying both of them together can separate the feature. In the future work, the classification model such as neural networks should be implemented to classify the fault kind automatically.

References

[1] P. Vas, Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines, Clarendron Press, Oxford, 1993.

[2] G. B. Kliman and J. Stein, "Induction motor fault detection via passive current monitoring," International Conference in Electrical Machines, Cambridge, MA, pp. 13-17, August 1990.

[3] K. Abbaszadeh, J. Milimonfared, M. Haji, and H. A. Toliyat, "Broken Bar Detection In Induction Motor via Wavelet Transformation," IECON'01: The 27^{th} Annual Conference of the IEEE Industrial Electronics Society, pp. 95-99, 2001.

[4] Masoud Haji and Hamid A. Toliyat, "Pattern Recognition-A Technique for Induction Machines Rotor Fault Detection Eccentricity and Broken Bar Fault," Conference Record of the 2001 IEEE Industry Applications Conference, vol. 3, pp. 1572-1578, 30 Sept.-4 Oct. 2001.

[5] S. Nandi, H. A. Toliyat, "Condition Monitoring and Fault Diagnosis of Electrical Machines – A Review," IEEE Industry Applications Conference, vol. 1, pp. 197-204, 1999.

[6] B. Yazici, G. B. Kliman, "An Adaptive Statistical Time-Frequency Method for Detection of Broken Bars and Bearing Faults in Motors Using Stator Current," IEEE Trans. On Industry Appl., vol. 35, no. 2, pp. 442-452, March/April 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Interactive musical editing system for supporting human errors and offering personal preferences for an automatic piano

— Databases and way of automatic translating —

Eiji Hayashi
Mechanical Engineering System
Kyusyuu Institute of Technology
680-4,Kawatu, Iizuka City,
Fukuoka pref.,820-8502

Yasuaya Takamatu
Mechanical Engineering System
Kyusyuu Institute of Technology
680-4,Kawatu, Iizuka City,
Fukuoka pref.,820-8502

and Hajime Mori
Apex Corporation
2-3-14, Kudanminami,
Chiyoda-ku, Tokyo, 102-0074

Abstract

Soft wares for computer music generally can't create performance information that will simulate a person's ability to play a new piece of music by sight, because even for a short piece of music is comprised of 1000 or more notes. As well, a large amount of time is required for a user to arrange the tempo, dynamics and so on in a reproducing piece of music. Therefore, we have developed a interactive musical editing system to help avoid human errors and that allows the expression of personal preferences in composing for an automatic piano. This system has been developed so that the user can edit performance information efficiently and reproduce at sight of a new piece of music. Three databases are created based on the music theory, e.g., music grammar, musical reinterpretation and the user's preference. A method of translating the musical data using these databases is described.

Key words: Automatic piano, Knowledge data base, Intelligence, Music interface, Computer music

1. Introduction

Recently, great progress has been made in the development of electronic devices, electronic instruments, and hard wares for computer music. However, such devices still cannot produce the sound of acoustic instrument with a high degree of accuracy. In light of this, we have developed an automatic piano aimed at imitating an actual piano [1].

The soft ware for computer music requires inputting data that includes tempo, dynamics and so on. But because there are 1000 or more notes even for a short piece of classic music, it is difficult for the user to edit performance information by sight, and to accurately reproduce the musical piece. As well, the music user must edit a new piece of computer music from the beginning. But musicians can play a new piece of music by sight because human generally holds memorizing and learning from previous experiences. Computer music users hold memorizing and learning on operations and editing way but soft wares are not memorizing and learning the user's playing way and the habit of user's performance. This requires that a new piece of computer music be edited from the beginning.

Therefore, we have developed a soft ware for the automatic piano that we have developed and the computer music that be possible to memorize and learn the users' habits in the performance information [2][3].

This paper describes the architecture of the system, the databases that structure to be edited efficiently, and the method for translating the musical data.

2. System

A block of the system is shown in Figure 1. The user basically edits performance information and operates the system through a user interface GUI. A user interface shown in Figure 2 has been constructed so that the user can efficiently edit the performance information. In the user interface, a piano score and graphs desicribing the performance information are given.

The performance information contains the note and pedal information shown in Tables 1 and 2. The note information consists of 6 parameters that include pitch (Key), dynamics (Velo), intervals between notes (Step),

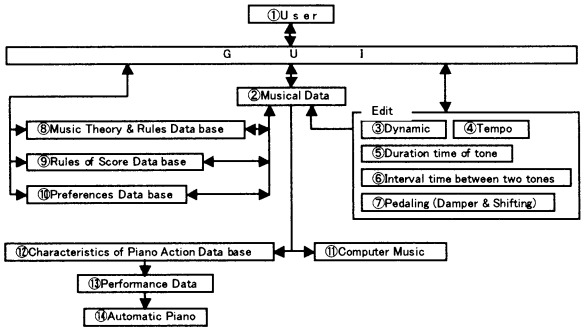


Figure 1 Block of system

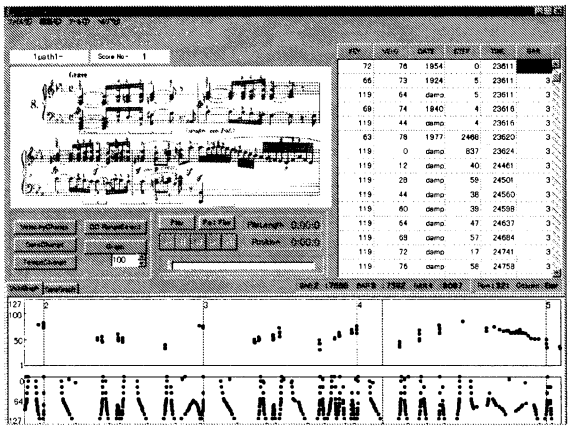


Figure 2 User interface

duration (Gate), tempo (Time), and bar number (Bar). The tempo, dynamics and duration of tone can change by respective desired ratios or as a whole and a group. Pedal information consists of the kind of pedals (Key), depth of pedal position (Velo), the number (time) that the pedal is stepped on, and bar number (Bar).

The databases can also be edited voluntarily, and the performance information can be changed automatically using databases.

The performance information edited this way can be reproduced using our automatic piano and a computer music system.

3. Databases

A piano performance is based fundamentally on the musical notation indicated on the score, including the notes. Furthermore, the expression of a musical piece differs according to the performer. This difference is base on various elements, such as tempo, the dynamics, the duration of each note, and the use of the pedals. And we have

Table 1 Performance information

Parameter	Reference	Unit
Key	Key No.21 - 108	
Velo	Dynamics 0 -127	
Gate	Duration of tone	ms
Step	Interval time between tones	ms
Time	Producing tone of timing	ms
Bar	Bar number	

Table 2 Pedal infprmation

Parameter	Reference	Unit
Key	Damper(No.109) Shift(No.110)	
Velo	Position of pedal 0 - 127	
Gate	-	
Step	-	
time	Positioning time	

thought that such differences are a pianist's feature. Such difference is unique to each performer. Therefore, we have consisted of 3 databases: a musical rules' database regarding the know-ledge of the music grammar, a score database regarding the music reinterpretation and the user's preference database.

The structures of those databases and a system with their databases are mentioned in the sections below.

3.1 Musical ules Database

Various kinds of musical notations and musical terms besides notes are contained in a piano score intended to direct the performance in regard to dynamics and other modes of interpretation. It is important for a user to acquire complete information regarding musical signs, so we divided the musical rules database into 5 groups. The structure of the musical database is shown in Table. 3.

3.2 Score Database

Pianists generally acquire various kinds of information regarding a particular piece of piano music. For example, they recognize a distinction between the right and the left hands and the respective roles of the melody and accompaniment, and use this information to give expression to their feelings through the music. Therefore, we have created a score database as shown in Table 4. This database consists of the 3 groups shown below.

Element data mainly gives distinction and role between hands regarding a note. General performance data has difficulty in distinguishing which hand is used in a piano

Table 3 Musical database

Parameter	Reference
Dynamics	*mf*, *pp*, *fp* etc
Alteration	*cresc.*, *rit.* etc
Way of Playing	*accent* *staccate* etc
Rhythm	1/2 ,3/4 etc(including stress)
Tempo	*Allegro* etc

Table 4 Score database

Parameter	Reference
Element	The role of a note ex. Right or Left hand
Music sign	Position of symbol Effect range
Same	Repeat sign etc

performance. Hence, this distinction needs to be a clear as possible. This data is written where the note sign is placed within a chord.

Note sign data indicates where a note sign is placed, and how long that note should be played. Because a performance always is advanced backward of a score as pianists consider a previous influence of the performance.

A same data mainly gives the same kind of phrases and repeat of the musical sign with frequency. Interval, tempo and dynamics in the same kind of phrases are different slightly, but the phrases have a lot in common form of performance.

Thus, if a user once edited phrases the frequently appears in the piece, all of such phrases throughout the piece are altered automatically. The user can then make smaller change to other passage.

3.3 Preference Database

A performance of pianist has a specific characteristic of a piano performance is an expression of a particular pianist. Likewise, if a user creates a work of music using a computer, the performance should also be an expression of the user. However, there is a decisive difference between a pianist and a user using a computer. Pianists improve their skillswith performance, and they can perform an unseen and an familiar piece of music by sight based on their experiences and know ledges, but a user can't do so. Because the user can't rewrite all the performance data to lots of notes by sight. Therefore, a preference database shown in Table 5 is structured based on the users' characteristic and to allow more efficient editing

Table 5 Preference database

Parameter	Reference
Note vector	Vector based on Velo and Step
Phrase Data	Note vector on a phrase
Gate ratio	Gate ratio on a way of playing

of the performance data regarding a new piece of music.

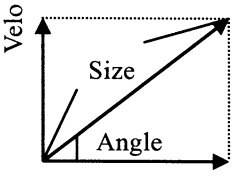


Figure 3 Note vector

The preference database consists of the note vector described below that can be extractedfrom performance data when a user is editing a piece of music. As well, the preference database is divided in 3 data regions, e.g., note vector, phrase and gate ratio of note that has musical sign. The note vector consists of elements of the velo and step elements shown in Figure 3.

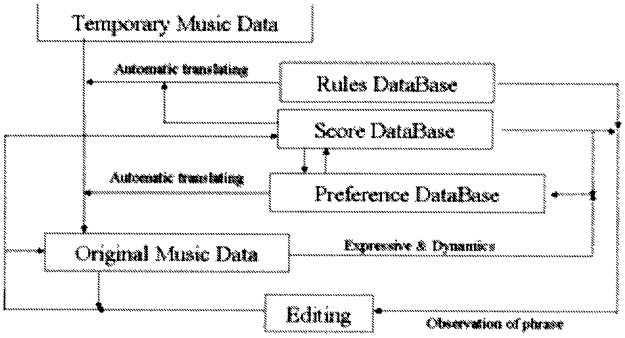


Figure 4 Way of translation

4. Translating for supporting edit

It is difficult for a user to give expression to temporary performance data (TPD) by editing, because the user can't get impression of performance as CD when the TPD was reproduced. So, if user edited the TPD, editing the TPD wiout expression information would require a great deal of time. Therefore, a way of translating this information for editing is showed in Figure 4.

Here a part of the first movement of Beethoven's Sonata No. 8 shown in figure 5 is used.

By translating the TPD shown in Figure 6 including the dynamics, rhythm and tempo of the musical rules database, we arrived at Figures 7-9, respectively.

Then, using elements of the score database, we arrived at figure 10. After the completion of all the processes

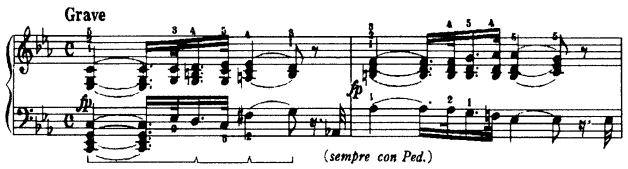


Figure 5 First movement of Beethoven's Sonata No. 8

described above, the original music data shown in figure 11 was made, and becomes allowing the user to edit this piece of music.

The results of translation using the preference database are showed in Figures 12 -13. Figure 12 shows the results of translation that changed velo and step in the original music data as a size of the note vector held in the preference database. Figure 13 shows the results of translation that changed velo and step as the direction of note vector held in the preference database.

5. Conclusions

We have developed 3 databases and a way of translation process for supporting the editing of a piece of computer music on our automatic piano and computer music. The results show that the system could offer users a performance data containing musical expression, making it easier for a user to edit performance data.

References

[1] Hayashi, E. et al, "Behavior of piano-action in a grand piano. I ", Journal of acoustical Society of America, Vol.105, pp.3534-3544, 1999.

[2] Asami,K and Hayashi,E.et al, "An intelligent supporting system for editing music based on grouping analysis in automatic piano", IEEE Proceedings of RO-MAN'98, 672-7, 1998.

[3] Hayashi,E et al, "Interactive musical editing system for supporting human errors and offering personal preferences for an automatic piano", AROB 7th, pp.513-516, 2002.

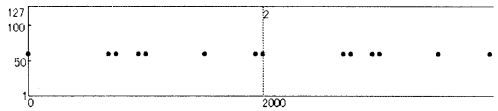


Figure 6 Temporary Music Data

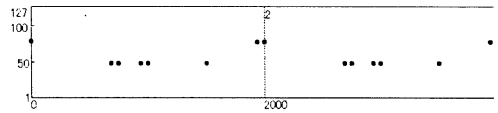


Figure 7 Result of Dynamics signs

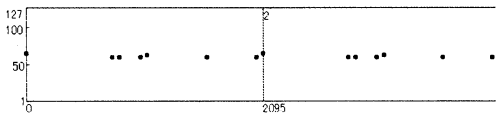


Figure 8 Result of Rhythm

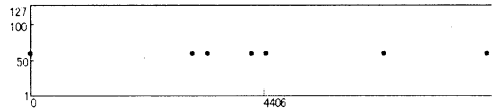


Figure 9 Result of Temp

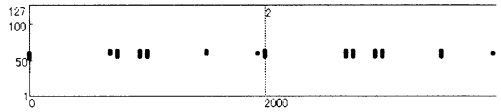


Figure 10 Result of Melody

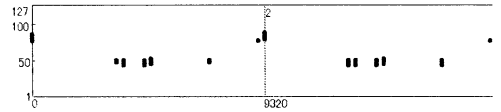
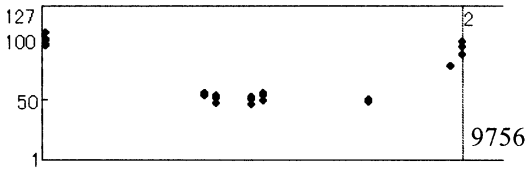
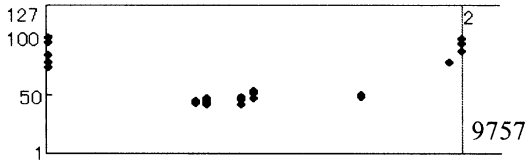


Figure 11 Result of all processes

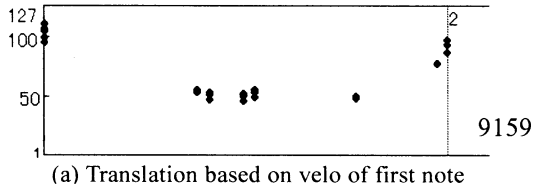


(a)Translation based on velo of first note

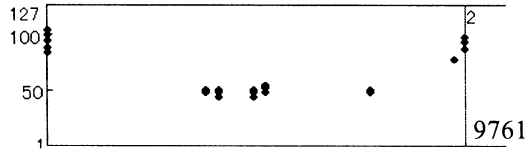


(b) Translation based on step of first note

Figure 12 Translation results when dize of note vector was held



(a) Translation based on velo of first note



(b) Translation based on step of first note

Figure 13 Translation results when directionof note vector was held

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Developing Potential Robots to Facilitate Humanitarian Demining: Challenges, Needs and Design Requirements

Maki K. Habib
School of Engineering and Science, Monash University
Bandar Sunway, Selangor, Malaysia
maki@ieee.org

Abstract

The fundamental goals of humanitarian demining are to detect and clear mines from infected areas efficiently, economically, reliably, rapidly and as safely as possible in order to make these areas economically viable and useful for development. Accurate and reliable mine detection techniques and technologies capable of area detection and clearance are crucial for successful demining. There is still the constrain on the resources to speed up the demining process in terms of time, cost, and personnel along with the issue of safety. A portable handheld approach to sensor movement is slow and hazardous for the individual but it is still needed in difficult or physically constrained environments. Mechanized and robotized solutions properly sized with suitable modularized structure and well adapted to local conditions of minefields can greatly improve the safety of personnel as well as efficiency and flexibility. Such robots can be equipped with mine detector and work to locate and mark each mine automatically. This paper introduces the problem of landmines and their impact. It focuses on the difficulties and requirements of solutions in the context of humanitarian demining.

Keywords: Humanitarian Demining, Mine detection, Demining Robot, Landmine, Mobile Robot.

1 Introduction

Locating, and removing through destruction or neutralization of all forms of dangerous battlefield debris, particularly landmines are vital prerequisites for any region to recover from their impact. This involves a great deal of effort and time, and large risk, which results in high clearance cost per surface unit. Many varied prerequisites have to be observed, such as soil and type of topology, as well as type of contamination [1-6]
The major effect of mines is to deny access to land and its resources, causing deprivation and social problems among the affected populations. In addition the medical, social, economic, and environmental consequences are immense. The international United Nation (UNDHA) assesses that there are more than 100 million mines that pose significant hazards in more than 68 countries around the world. There exist many types of mines around the world; including more than 650 types of antipersonnel landmines [1, 2]. Anti-personal mines are harmful because of their unknown position and for the variety in explosive load, the activation means, action range, and the effect on human bodies. Landmine technology ranges from very simple to high technology devices. Pressure, tripwires, tension or pressure release, electro-magnetic influence, and seismic signals can detonate mines. Some landmines are "hardened" against neutralization by explosives and other landmines have anti-disturbance mechanisms. In addition, mines may have a booster charge to enhance the power released by the detonator to a level that's enough to initiate the main charge. Mines may have been in place for many years, they may be corroded, waterlogged, impregnated with mud or dirt, and can behave quite unpredictable. Some mines were buried too deep to stop more organized forces finding them with metal detectors. Deeper mines may not detonate when the ground is hard, but later rain may soften the ground to the point where even a child's footstep will set them off. Modern landmines are fabricated from sophisticated non-metallic materials and incorporate advance electronics. They are harmful because of their unknown position and are often difficult to detect.

2 The Military Demining Approach and Humanitarian Demining Requirements

Humanitarian demining scenarios differ from military ones in many respects. The objectives and philosophy are different. Solutions developed for the military are generally not suitable for humanitarian demining. The military use the term 'breaching' to describe their main mine-clearing concern. It is dictated by the strategies of warfare, tactical countermining comprises operations that allow an attacking force to penetrate or avoid mines rapidly as it attacks a target or to speedily clear areas to sustain specific operations. The pace is very quick. Individual mines need not be found, and casualties from mines and other weapons are expected and accepted. Significant resources can be brought while not all the

mines need to be cleared and any clearance rate over 80% is generally considered satisfactory. Also, the military accepts relatively high risk that some of their vehicles and soldiers will still be destroyed and killed even after breaching has been completed. The time is a critical factor in military breaching. Military mine clearance equipment tends to be expensive and may be high-tech, large in size, requiring highly trained logistical personnel. The means to do so is by mechanical landmine clearance, such as, ploughs, flails rollers, tracks, etc. Mechanical landmine clearance approach is fast but it cannot assure safety and it is environmentally not friendly. With this technique, antipersonnel mines may be pushed on side or buried deeper or partly damaged making them more dangerous. The humanitarian situation is quite different. Humanitarian demining is by definition a total clearance of the land from all types of mines. It is carried out in a post-conflict context, and its objective is to decontaminate populated areas and infrastructure of all landmines, make land safer for daily living and restoration to what it was prior to the hostilities and to allow people to use their land without fear. The amount of time it takes to clear an area is less important than the safety of the clearance personnel. Humanitarian demining requires that the entire land area be free of mines and hence the need to detect, locates, and removes reliably every single mine from a targeted ground. Safety is of utmost importance, and casualties are unacceptable. The standard to which clearance must be achieved is extremely high to make sure that there is not a mine being left in the ground before letting the people stepping in again. Any system to be developed should compliment this effort, not to hamper it or simply move the problem elsewhere. The risks to those carrying out the task must also be maintained at a lower level than might be acceptable in a military situation. Another consideration by humanitarian demining is the use of land for development, i.e., reducing the environmental impact that may results from the demining operation. Currently available technologies are not suited to achieve the objectives of humanitarian demining. What is required is to have a landmine detection system that is almost 100% effective, low cost, fast, and easy to use.

3 Humanitarian Mine Clearance and the Difficulties

The development of new demining technologies is difficult because of the tremendous diversity of environmental conditions in which mines are laid and because of the wide variety of landmines. The environmental conditions whereas mines are often laid, cover weather (hot, humid, rainy, cold, windy), the density of vegetation (heavy, medium, small, none), type of soil (soft, sand, cultivated, hard clay, covered by snow, covered with water), and type of the terrain (rocky, rolling, flat, desert, beaches, hillside, muddy, river, canal bank, forest, trench). Also, residential, industrial and agriculture area, each has its own features and needs to be considered. Landmines are many in terms of type and size. The size ranges from small enough to fit into a child's hand, and weighting as little as 50 grams, to large antitank mines. Antipersonnel mines come in all shapes and colors are made from a variety of materials, metallic and nonmetallic. Metal detector works well with metal cased mines, but metal in modern mines has been increasingly replaced by plastic and wood. New mines will soon be undetectable by their metallic content. Anti personnel mine can be laid anywhere and can be set off in a number of ways because the activation mechanisms for these mines are not the same. Activation methods can be classified into three categories, pressure, electronic, and command detonation (remote control). Mines may have been in place for many years, they may be corroded, waterlogged, impregnated with mud or dirt, and can behave quite unpredictable. Some mines were buried too deep to stop more organized forces finding them with metal detectors. Deeper mines may not detonate when the ground is hard, but later rain may soften the ground to the point where even a child's footstep will set them off. Trip wires may be caught up in overgrown bushes, grass or roots. In addition, there is no accurate estimate on the size of the contaminated land and the number of mines laid in it. This tremendous diversity combined with the fact that existing equipment and techniques make demining slow, dangerous and man-intensive makes the elimination of landmines and extremely complex mission. Improvements in technology are critical to the success of efforts intended to reduce this threat.

The diversity of the mine threat points to the need for different types of sensors and equipment to detect and neutralize landmines. The requirements to develop equipment for use by deminers with different tra ining levels, cultures, and education levels greatly add to the challenge. The solution to this problem is very difficult from a scientific and technical point of view because, given the nature of landmines and the requirements of humanitarian demining, any instrument must be 100% reliable for the safety of the operators and the people whom will use the land.

4 Role of Robotics in Humanitarian Demining

A portable handheld detection approach to sensor movement is slow and hazardous for the individual deminers but it is still needed in difficult and physically constraint environments. Armored vehicles may not thoroughly protect the occupants and may be of only

limited usefulness in off-road operations. To increase mine clearance daily performance by improving productivity, accuracy, and to increase safety of demining operations and personnel, there is a need for an efficient, reliable and cost effective humanitarian mine action equipment with mobility and some level of decision making capability. Such equipment to be equipped with selectable sets of mine detectors and work to locate and mark individual mines precisely, and for a later stage to neutralize the detected mines. Robotization of humanitarian demining reflects the use of teleoperated, semi-autonomous or autonomous robots with mobile platform. Robotics solutions properly sized with suitable modularized mechanized structure and well adapted to local conditions of minefields can greatly improve the safety of personnel as well as work efficiency, productivity and flexibility. Robots can speed the clearance process when used in combination with handheld mine detection tools, and they may also be useful in quickly verifying that an area is clear of landmines so that manual cleaners can concentrate on those areas that are most likely to be infested. Also, solving this problem presents challenges in robotics research field. Robotics research requires the successful integration of a number of disparate technologies that need to have a focus to develop,

i. Flexible mechanism and modular structure
ii. Mobility, intelligence and behavior based control
iii. Human support functionalities and interaction.
iv. Sensors integration and data fusion.
v. Different aspect of autonomous or semi -autonomous navigation.
vi. Planning, coordination, and cooperation among multi robots.
vii. Wireless connectivity (inter-process and inter-robot communication) and natural interaction with human.
viii. Virtual reality and real time interaction to support planning and logistics of robot service.

The possible introduction of robots into demining process can be done through surface preparation and marking, speeding-up detection, and mine removal or neutralization. It is clear that it is difficult to design a universal robot/machine to meet demining requirements. Furthermore, the use of many robots working and coordinating their movement will improve the productivity of overall mine detection process through the use of team cooperation and coordination.

It is absolutely clear that in many cases the environment to be dealt with is so hostile that no autonomous robot has little chance to be used. Tele-operated are promising but are limited too, because their remote human controllers would have limited feedback and would be unable to drive them effectively in real time. Strangely enough, this is particularly true for urban areas normally full of rubble, while agricultural areas seem to be better, but that is not always true.

Difficulties can recognize in achieving a robot with specifications that can fulfill the illustrated requirements. In spite of this, many efforts have been recognized to develop an effective robot for the purpose to offer cheap and fast solution. Such efforts include three wheels Dervish robot (UK), Pemex and Shrimp (EPFL, Switzerland), Ariel (IS Robotics, Canada), TRIDEM and AMRU-1, 2, and 4 (Belgian), legged robot, Track hunter (Japan), etc. [3-10].

5 The Requirements for Designing Efficient Demining Robots

Research into individual, mine-seeking robots is in the early stages with numerous institutions and companies involved. In their current status, they are not an appropriate solution for mine clearance. This is because, their use is linked with sensing improvements, the difficulties facing any automated solution due to the variety of mines and minefields, a variety of terrain in which mine can be found. Also, currently robot solutions are too expensive to be used for humanitarian demining operations in countries like Angola, Afghanistan and Cambodia.

The development of a unique robot that can operate under wide and different conditions is not a simple task. In the short term, it appears that the best use of robotics will be as vehicles with arrays of detection sensors and area mine clearance devices. A possible idea in using robots for demining is to design a series of simple and modularized robots, each one capable of performing one of the elementary operations that are required to effectively clear a minefield. An appropriate mix of such machines should be chosen for each demining task, keeping in mind that it is very unlikely that the whole process can be made fully automatic. The effort devoted to robotic solutions would be more helpful if it were directed at simple equipment improvements and low-cost robotic devices might provide some useful improvements in safety and cost-effectiveness in the short to medium term. Before the robot can do its detection task, there is a need to employ mechanized systems that are able to remove obstructions that deter manual and canine search methods without severely disturbing soil. The practical difficulties in using robots for mine clearance are many. Before applying robotics technology for mine clearance process, it is necessary to specify the basic requirements and specifications for the robot to have to achieve a better performance. These requirements include the mechanism, algorithms, functions and use.

a. It is essential to design a robot that will not easily detonate any mines it might cross on its way (good

resistance to explosions). Also, the static ground pressure and impact force due to the weight of the robot should be low enough not to set off any mine.

b. The robot should be able to cross safely over the various ground conditions. This can be achieved by having adaptable and modular mechanism both for the mobility and structure.
c. The robot must be, low purchased cost and cheap to run, small, lightweight, and reliable.
d. It should be portable, easy to transport and deploy.
e. The robot should be highly mobile, and the mobility should ensure to overcome the average obstacles and ease of moving around.
f. It should employ multi sensorial system for detecting and recognizing mine.
g. It should have a suitable mechanism for self-recovery for some levels of the problems that may face during it work.
h. Consideration should be given to have a robot that can resist water, sand, temperature and humidity.
i. The mechanical design should be as simple and low tech so that anyone can make it from a local material, such as, bicycle components, bamboo, etc.
j. The robot has to work in more than one operational mode such as, autonomous, semi-autonomous and teleoperated modes while keeping the deminer out of physical contacts with mines.
k. It should be capable of withstanding explosive blast without suffering major damage. At the minimum the high tech parts of the robot that cannot be replaced locally should be well protected.
l. The robot should be suitable for maintenance and repair by indigenous users. Ease of maintenance is built in at the design stage so that if repair is ever necessary it may be carried out locally without the use of special test equipment.
m. Sustaining a reasonable power supply to enable the robot to operate for long period.
n. The robots can be tested and deployed without expensive environment and enormous launch cost.
o. It should have efficient navigation system with proper localization of sensors on the minefield. Good man-machine-interfaces with ergonomics of lightweight portable control stations, user-friendly interaction capability.
p. The robot should deploy low cost and practical technology. Wherever possible, robot's chosen components can be found or manufactured in the mine-affected region.

6 Conclusions

It is clear that it is difficult to realize a universal robot/machine design with locomotion capability that can work under all types of terrain along with weather and environmental constraint. Difficulties can be recognized in achieving a robot with specifications and functionalities that can fulfill the illustrated requirements. A modularized mechanical design is the best way to enable reconfiguring robot mechanical structure to suite humanitarian demining requirements. In the short and mid terms, robots can help to accelerate searching and marking mines. Also, It can be helpful to assurance quality control stage requirements for verification purposes. High cost and high tech features are additional constraints in using robots for demining. A Tool Box approach has to be adopted, and different procedures and technologies have to be used for the clearance of different types of land under different conditions.

References

[1] T. J. O'Malley, "Seek and Destroy - Clearing Mined Land", Armada International, pp 6-15, 1993.

[2] M. K. Habib, "New Potential and Development of Mine Detection and Sensing Technologies in the Context of Humanitarian Demining", 5th International Conf. on Mechatronics Technology ICM'2001, Singapore, 2001.

[3] S.H. Salter, "Dervish: a Purpose-Designed Vehicle for Post War Detonation of Anti-Personnel Mines", internal report, 1995.

[4] J. D. Nicoud and M. K. Habib, "PEmex-B Autonomous Demining Robots: Perception and Navigation Strategies", IROS'95, pp. 419-424, 1995.

[5] J. D. Nicoud, Ph. Machler, "Robots for Anti-Personnel Mine Search", Control Engineering Practice, 4 (4), pp. 493-498, 1996.

[6] Y. Baudoin, M. Acheroy, M. Piette, and J.P. Salmon, "Humanitarian Demining and Robotics ", Mine Action Information Center Journal, Vol. 3, No. 2, Summer 1999.

[7] S. Havlík and P. Licko, "Humanitarian Demining: The Challenge for Robotic Research", The Journal of Humanitarian Demining, Vol. 2, No.2, May 1998.

[8] K. Nonami and Q. Huang, "Humanitarian Mine Detection Six-Legged Walking Robot COMET-II with Two Manipulators", Proceedings of 4th CALWAR-2001, pp. 989-996, 2001.

[9] Dawson-Howe, K. M. and T. G. Williams. "The Detection of Buried Landmines Using Probing Robots." Robotics and Autonomous Systems, Vol. 23, No. 4, pp. 235-243, 1998.

[10] G. Muscato and G. Nunnari, "Leg or Wheels? WHEELEG a hybrid solution", CLAWAR'99 International conference on climbing and Walking Robots, Portsmouth, U.K, September 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Questionnaire Results of Subjective Evaluation of Seal Robot at the Japan Cultural Institute in Rome, Italy

Takanori Shibata [1,2], Kazuyoshi Wada [1,3], and Kazuo Tanie [1,3]

1 National Institute of Advanced Industrial Science and Technology
1-1-1 Umezono, Tsukuba 305-8568, Japan
(Tel. +81-29-861-7299, E-mail: {shibata-takanori, k-wada, tanie.k}@aist.go.jp)

2 PERESTO, JST

3 Institute of Engineering Mechanics, University of Tsukuba

Abstract: The seal type robot, Paro has been researched and developed as an example of mental commit robot that provides psychological, physiological, and social effects to human beings through physical interaction. In the previous research, we introduced Paro at Tokyo Big Sight in Tokyo, Japan, and evaluated them by questionnaires to many visitors. The results showed that Paro was accepted many Japanese people. Moreover, feeling of touch and appearance of the robot were key points to evaluate mental commit robot in Japan. In this paper, 95 subjects evaluated the seal type robot, Paro by questionnaires in an exhibition at the Japan cultural institute in Rome, Italy for 4 days from June 25th to 28th, 2003. This paper reports the results of statistical analysis of evaluation data.

1. Introduction

We studied and developed artificial emotional creatures as the examples of robots that coexist with people [1-8]. The artificial emotional creatures exist as subsidiaries in everyday life similar to pets, hold equal relationships with people, move by orders from people and act autonomously, and contact with people [9], giving people pleasure and relaxation. They prevent mental illness by enriching and healing the humans' heart, give pleasure to daily leisure, and commit themselves to the humans' mental part. They are referred to as "mental commit robots".

Seal type mental commit robot was applied to therapy of children at pediatric hospital and assisting activity of elderly people at elderly institutions [3-5]. The results showed that there were psychological merit, physiological merit, and social merit by interaction with the seal robot.

A study was conducted by questionnaires in "Exhibition of Dream Technology" held for about 3 weeks from July to August 2000, on how people subjectively evaluate robots in human interaction, a characteristic of mental commit robots [6]. Seal robots were used for evaluation. Respondents were exhibition visitors, and after demonstrators explained the purpose and functions of the robot, they were requested to interact with the robot. Multiple questions about the robot were asked by questionnaires. The tabulation results of subjective evaluation provided high evaluation on the whole. In addition, important factors to be key points in robot evaluation were extracted from the results of principal component analysis, thus confirming relationship between them and factors such as shape and the feel emphasized in design. Respondents were grouped to find differences in evaluation among groups. These results revealed that the evaluation factors of "favorable impression by contacting" and "favorable impression by appearance" are important as those pertaining to evaluation in the design and production of mental commit robots to be highly appreciated in subjective evaluation.

This time, we obtained an opportunity to exhibit the seal robots at the Japan cultural institute in Rome, Italy for 4 days from June 25th to 28th, 2003. The same study was conducted by questionnaires in the institute for 4 days. This paper reports results of statistical analysis of evaluation data.

The developmental purpose and process of the mental commit robot are stated in Chapter 2. The method of subjective evaluation is explained in Chapter 3, statistical analysis results of evaluation data are discussed in Chapter 4, and they are summarized in Chapter 5.

2. Seal Robot: Paro

2.1. Mental Commit Robot

Mental commit robots are intended to offer people not physical work or service but mental effects such as pleasure and relaxation as personal robots. Robots act independently with purposes and motives while receiving stimulation from the environment as with living organisms. Actions manifested by interaction with people are interpreted by people as if robots have hearts and feelings.

2.2. Seal Robot "Paro"

The robot and major functions are shown in Fig.1. The appearance was designed with a harp seal baby as a model, and the surface is covered with pure white fur. A newly developed plane tactile sensor is inserted between the hard inside skeleton and fur to express the soft natural feel and to permit the measurement of human contact with the robot. The robot has the 4 senses of sight (light sensor), audition (determination of sound

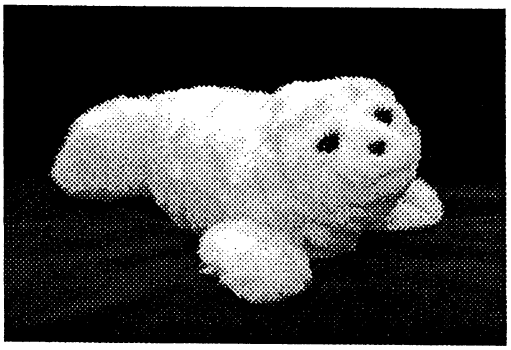


Fig. 1 Seal Robot: Paro

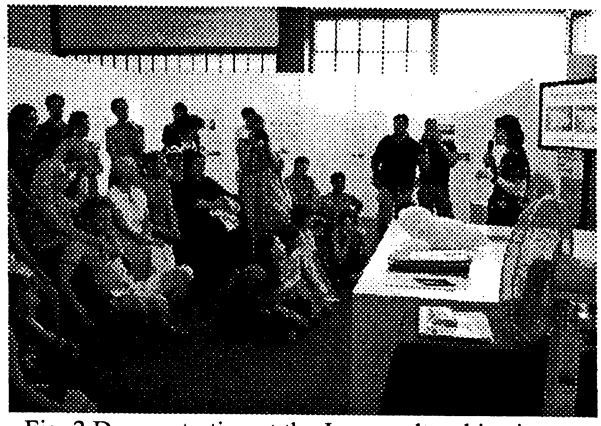


Fig. 2 Demonstration at the Japan cultural institute in Rome, Italy

source direction and speech recognition), and equilibrium in addition to the above-stated tactile sense. Mobile parts are as follows: vertical and horizontal neck movements; front and rear paddle movements; and each eyelid movement important as facial expression. The robot acts by using the 3 elements of the internal states of the robot, sense information from sensors, and daily rhythm (morning, daytime, and night) to manifest activities through interaction with people.

3. Purposes and Method of Subjective Evaluation

3.1. Purpose of Subjective Evaluation

In the previous studies, subjective evaluation was done with cat robots for 88 respondents and seal robots for 785 respondents in Japan. The objectives of this paper were that questionnaires from a lot of persons in Rome, Italy, were conducted on a large scale to collect subjective evaluation data about interaction with robots and analyze them statistically, thus analyzing and classifying the tendency of robot evaluation and elements as evaluation points. It was also intended to collect requests, opinions, and impressions to the development of mental commit robots that are used for future R&D.

3.2 Method of Subjective Evaluation

In Robotics exhibition hold at the Japan cultural institute in Rome, Italy for 4 days from June 25th to 28th, 2003, seal robots Paro Ver.7 was displayed. Demonstration was to be done through the whole period.

Table 1 Question about Subjects

Personal questionnaire
1. Sex ?
2. Age ?
3. Occupation ?
4. Do you like animals?
5. Are you owning any pets or do you want to own pets?
6. Why do you not have any pets?
7. Do you know about real baby seals?
8. How much free time do you have in daily life?

Table 2 Question about Subjective Evaluation of Paro

Evaluation questionnaire
1. Cute
2. Want to pet it
3. Want to talk to it
4. Has vitality
5. Easy to get friendly with
6. Has real expressions
7. Natural
8. Feels good to the touch
9. Fun to play with
10. Relaxing
11. Like
12. Needed in this world
13. Want it for myself

On the stage, demonstrators conducted one to three demonstrations a day. After the developmental process and purpose and the functions of mental commit robots etc. were explained for about 10 minutes, Paro was placed on the stage to allow the audience who visited the stage to touch the robot freely. A questionnaire was given to visitors who after interaction, and accepted this filled out the form. Total 95 questionnaires were obtained during the 4 days.

3.2.1. Questionnaire

The contents of the questionnaire are written in Italian, and classified largely into questions about the respondent (Table 1); and questions of 5-grade evaluation for subjective evaluation (Table 2). The answer time was different from person to person, but 5-10 minutes were necessary. Many children were included, so parents were requested to help them in answering questionnaires.

3.2.2. Condition of Demonstration (Interaction)

Paro was placed on the table to allow visitors to the stage to interact freely with it. The front of the stage was sometimes quite crowded and yet every cycle of demonstration took about 30 minutes to permit as many people as possible to experience interaction by turn.

Interaction mainly involved the actions of contacting and stroking, and women hugged it in some cases. In other cases, visitors called its name, or brought their faces close to it.

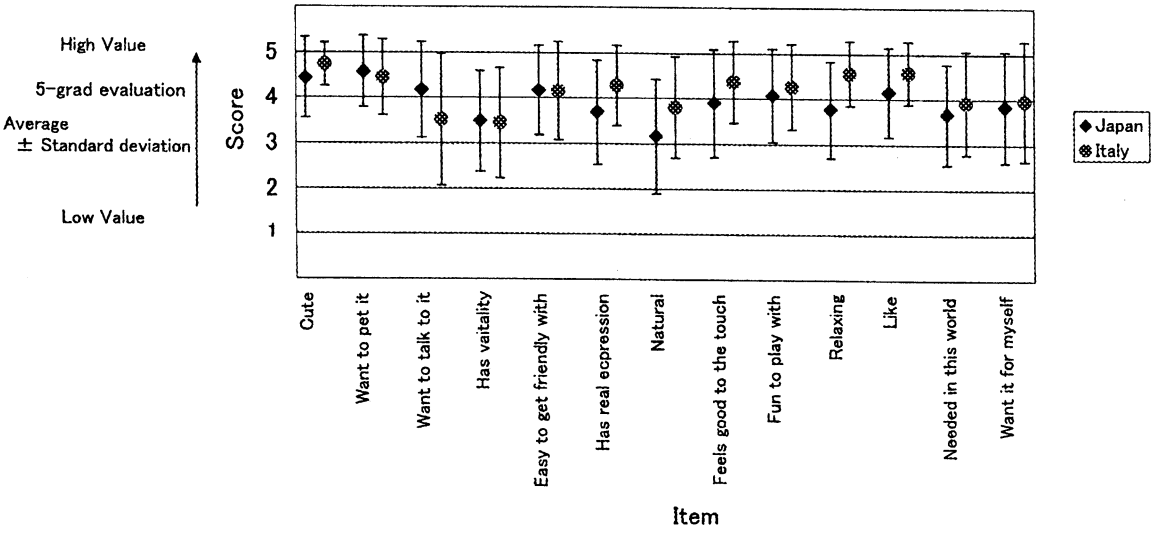


Fig. 3 Average and Standard Deviation of Answers to 13 Questions about Subjective Evaluation in Tokyo, Japan (641 sheets) and Rome, Italy (76 sheets)

4. Results and Discussion

4.1. Answer Results of Questionnaires

Fig.3 gives the values of average and standard deviation of answers to the 13 questions of subjective evaluation at the Japan cultural institute in Rome, Italy, and at the previous study in Tokyo, Japan. The results of subjective evaluation in Italy indicate that favorable answers were obtained for the most part, and the variables of "Cute," "Want to pet it," "Relaxing," and "Like" are highly evaluated. The results were very similar to those in the previous study in Tokyo, Japan.

4.2. Principal Component Analysis of Subjective Evaluation

Multivariate analysis was conducted to evaluate the answers of subjective evaluation comprehensively. In analysis, there is no external criterion and the contents of all 13 questions are explanatory variables for Paro evaluation; the multivariate analysis techniques of such variables include principal component analysis, factor analysis, and latent structure analysis.

Principal component analysis was used to do comprehensive evaluation by 13 explanatory variables, and because cumulative contribution becomes higher than the case of extracting the same number of factors by factor analysis. This analysis was conducted with 76 sheets consisting of complete answers to 13 questions.

4.3. Results and Discussion of Principal Component Analysis

The principal component analysis was done with 76 sheets of complete answers to 13 questions of subjective evaluation to extract three factors with eigen-value of 1 or higher. Kaiser normalization was done to enhance factor interpretation, and then three factors characterized by factor loading shown in Table 3 was obtained.

Table 3 Three Factors Characterized Factor Loading

	Three Factors		
Variable	Factor 1	Factor 2	Factor 3
Cute	0.742	-0.024	0.351
Want to pet it	0.586	0.494	0.014
Want to talk to it	0.484	0.696	0.090
Has vitality	0.309	0.516	0.457
Easy to get friendly with	0.172	0.207	0.724
Has real expressions	0.282	-0.022	0.747
Natural	0.399	0.287	0.688
Feels good to the touch	0.642	0.189	0.268
Fun to play with	0.752	0.313	0.182
Relaxing	0.606	0.087	0.435
Like	0.573	0.450	0.250
Needed in this world	-0.098	0.767	0.421
Want it for myself	0.214	0.855	0.000
%Total variance	25.359	46.775	65.805

4.3.1. Interpretation of Factors

The first factor is characterized by the variables of "Cute," "Feels good to the touch," "Fun to play with" and "Relaxing," and it can be interpreted as a factor of "favorable impression by contacting and appearance." The second factor is characterized by "Needed in this world" and "Want it for myself," and it can be interpreted as a factor of "necessity of Paro." The third factor is characterized by "Easy to get friendly with," "Has real expressions" and "Natural," and it can be interpreted as a factor of "impression like real animals."

4.3.2. Comparison of Factor Scores

To analyze factor scores in detail, respondents were grouped and average factor scores were compared between groups. Grouping was done with gender and age, and the factor score of each group was subjected to

Wilcoxon's test, a nonparametric test as a test for mean difference, because the normality or equal variance of each group cannot be expected. The comparison of factor score averages between proportions of respondent number of each group is shown in Fig.5.

In grouping by gender, no significant differences were seen in the scores for the three factors. In grouping by age, the second factor of "Necessity of Paro" was significant, and the group of less than 20 years old evaluated it highly. As for other factors, such as "Favorable impression by contacting and appearance" and "Impression like real animals," the group of less than 20 and over 50 years old tended to evaluate them highly.

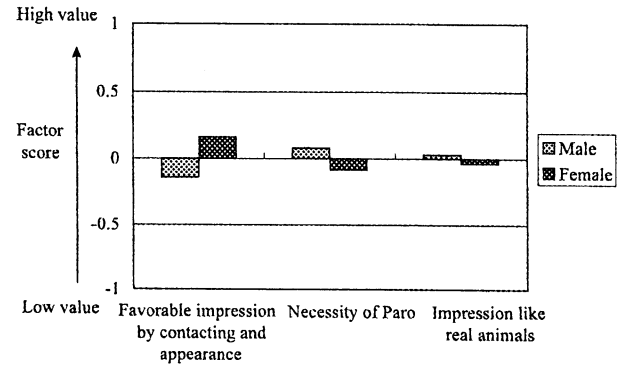
5. Conclusion

A seal robot, "Paro," was introduced to a lot of visitors in robot exhibition at the Japan cultural institute in Rome, Italy, and interaction with Paro was done to ask questionnaires about the subjective evaluation of Paro.

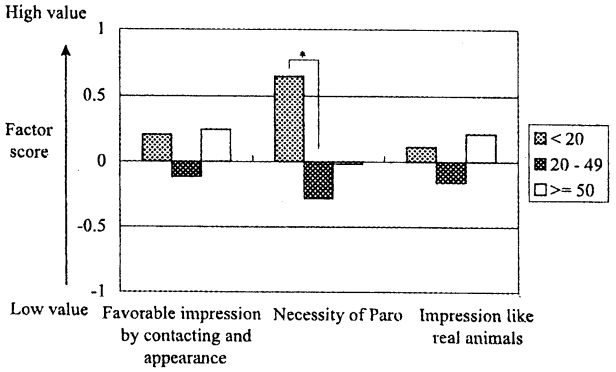
The tabulation results of subjective evaluation provided high evaluation on the whole, and were very similar in those of Japan. In addition, important factors to be key points in robot evaluation were extracted from the results of principal component analysis. Then, respondents were grouped to find differences in evaluation among groups. These results revealed that the evaluation factors of "favorable impression by contacting and appearance," "necessity of Paro" and "impression like real animals" are important in Italy, on the other hand, in Japan, "favorable impression by contacting" and "favorable impression by appearance" are important as pertaining to evaluation in the design and production of mental commit robots to be highly appreciated in subjective evaluation. Therefore, we should design mental commit robots to have real and natural expressions for Italian people. On the other hand, for Japanese people, deformations of the robots are important to have more favorable touch and appearance.

We conducted the same studies in U.K. (440 respondents) and Sweden (133 respondents) [7, 8]. Comparison between the results in Italy and those in the countries will be reported in the future.

These results as requests and impressions obtained in questionnaires will be used for the improvement of Paro.



(a) Comparison by Gender



(b) Comparison by Age

Fig. 5 Proportion of Number of Each Group Member and Comparison of Factor Scores Average

References:

[1] T. Shibata, et al., Emotional Robot for Intelligent System - Artificial Emotional Creature Project, Proc. of 5th IEEE Int'l Workshop on ROMAN, pp.466-471, 1996

[2] T. Shibata, et al., Emergence of Emotional Behavior through Physical Interaction between Human and Robot, Procs. of the 1999 IEEE Int'l Conf. on Robotics and Automation, 1999

[3] T. Shibata, et al., Mental Commit Robot and its Application to Therapy of Children, Proc. of the IEEE/ASME Int'l Conf. on AIM'01, paper number 182 and 6 pages in CD-ROM Proc., 2001

[4] K. Wada, et al., Robot Assisted Activity for Elderly People and Nurses at a Day Service Center, Proc. of the IEEE Int'l Conf. on Robotics and Automation , 2002

[5] T. Saito, et al., Examination of Change of Stress Reaction by Urinary Tests of Elderly before and after Introduction of Mental Commit Robot to an Elderly Institution, Proc. of AROB, 2002

[6] T. Shibata, et al., Subjective Evaluation of Seal Robot: Paro -Tabulation and Analysis of Questionnaire Results, Jour. of Robotics and Mechatronics, Vol. 14, No. 1, pp. 13-19, 2002

[7] T. Shibata, et al., Tabulation and Analysis of Questionnaire Results of Subjective Evaluation of Seal Robot at Science Museum in London, Proc. of the 2002 IEEE Int. Workshop on ROMAN, pp.23-28, 2002

[8] T. Shibata, et al., Subjective Evaluation of Seal Robot at the National Museum of Science and Technology in Stockholm, Proc. of the 2003 IEEE Int. Workshop on ROMAN, paper #397, 2003

[9] M. M. Baum, et al., Physiological Effects of Human/Companion Animal Bonding, Nursing Research, Vol. 33. No. 3, pp.126-129, 1984

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Intelligent Control System based on Artificial Life

Tu Xuyan Yin Yixin Zeng Guangping Ban Xiaojuan
Institute of Computer and Systems Science
University of Science and Technology Beijing
E-mail: tuxuyan@vip.sina.com

Abstract:

This paper proposes the concept and structure of the Intelligent Control System based on Artificial Life (ICS/AL). The Intelligent Controller based on Artificial Brain (IC/AB), The Feedback Measure based on Artificial Sense (FM/AS) and the Control Actuator based on Artificial Organ (CA/AO) are described. The scheme of an intelligent control system of the Humanoid Intelligent Robot (HIR) is given.

Key words:

Intelligent Control System, Artificial Life, Artificial Brain, Artificial Sense, Artificial Organ.

Introduction

Intelligent Control is the development based on the combination of Artificial Intelligence and Control Technology, that can be shown as follows:

AI + CT → IC

Where: AI — Artificial Intelligence, CT— Control Technology, IC — Intelligent Control, "+" — "Combination", "→"— "Development"

Artificial Life (AL) is the development and application of Artificial Intelligence (AI).

In this paper, the concept and structure of intelligent control system based on Artificial Life (ICS/AL) are proposed and discussed from the view point of Generalized Artificial Intelligence (GAI) and Generalized Artificial Life (GAL).

1. Concept of ICS/AL

The concept of Intelligent Control System based on Artificial Life (ICS/AL) is shown by the expression (1) as following:

ICS + AL → ICS/AL (1)

Where, ICS — Intelligent Control System, AL — Artificial Life, ICS/AL—Intelligent Control System based on Artificial Life, "+" — "Combination", "→"— "Development"

The expression (1) means that based on the combination of ICS and AL, the ICS /AL can be developed by methodology and technology of artificial life, for example: Artificial Brain (AB), Artificial Sense (AS) and Artificial Organ (AO).

The ICS/AL is the new ICS based on AL, It has the humanoid (or animaloid) function and structure, namely, the ICS/AL is the simulation, extension or expansion of biological ICS of human (or animal).

2. Structure of ICS/AL

The basic structure of intelligent control system based on Artificial Life (ICS/AL) is shown as Fig1.

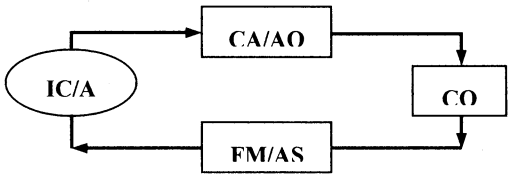


Fig1: Basic Structure of ICS/AL

In Fig1: IC/AB— Intelligent controller based on Artificial Brain, CA/AO — Control Actuator based on Artificial Organ, FM/AS — Feedback measure based on Artificial Sense, CO — Controlled Object

The basic ICS/AL is a close-loop feedback control System. In general, The ICS/AL could be various structures. For example:

Complex ICS/AL, It have the complex structure with the combination of "Open loop + Close loop" and "Frontfeed + Feedback", It is similar to the physiological parameter complex control system in human body.

Multilevel ICS/AL, It have the hierarchical structure with the combination of "Centralized control + Decentralized control", "Whole control + Local control", It is similar to the human multilevel neural control systems.

Harmonic ICS/AL It have the structure with the combination of "Fast control + Slow control" and "Function control + Region control", It is similar to the "Neural control + Hormonal control" harmonic control system in human body.

3. Intelligent Controller IC/AB

The concept of Intelligent Controller based on Artificial Brain is shown as expression (2):

$$\mathbf{IC + AB \rightarrow IC/AB} \qquad (2)$$

Where: IC — Intelligent controller, AB— Artificial Brain, IC/AB — Intelligent Controller based on Artificial Brain., "+" — "Combination", "→"— "Development"

The expression (2) means that based on combination of methodology and technology of IC and AB, We can develop new intelligent controller IC/AB.

The Artificial Brain (AB) is a technical model of human (or animal) brain, it has the 'Artificial Thinking Intelligence' (ATI), The ATI is the simulation, extension and expansion of human (or animal) thinking intelligence. For examples:

Artificial Neural Network (ANN) is some Local or partial artificial brain, **Expert system (ES)** is the functional simulations of thinking intelligence of human expert, **Artificial cerebellum (AC)** is the model of human cerebellum with the function of control and coordination of motion, **Artificial Left Brain (ALB)** has the artificial logical thinking intelligence, **Artificial Right Brain (ARB)** has the artificial formal thinking intelligence, **Artificial Whole Brain (AWB)** has the artificial logical and formal thinking intelligence.

By means of IC/AB, some high level intelligent controller can be achieved. Such as:

Self — Harmonizing Controller (SHC), The Self —Harmonizing Controller can achieved automatically the coordination of "Fast Control + Slow Control", "Region Control + Function Control", and It is simulated to "Neural Control + Hormonal Control" harmonizing controller in human body.

Self — Associating Controller (SAC), The association of both artificial formal thinking and logical thinking can be achieved automatically by the humanoid self-associating controller. It is useful to solve some complex control problem in uncertain or non-determinate environment.

4. Feedback Measure FM/AS

The concept of Feedback Measure based on Artificial Sense (FM/AS) is shown as expression (3):

$$\mathbf{FM + AS \rightarrow FM/AS} \qquad (3)$$

Where: FM — Feedback Measure, AS — Artificial Sense, FM/AS — Feedback Measure based on Artificial Sense, "+" — "Combination", "→"— "Development"

The expression (3) means that based on combination of methodology and technology of FM and AS, We can develop new feedback measure FM/AS.

The Artificial Sense is the simulation, extension or expansion of human sense (or other animal sense). For example:

Artificial Eyes, It is a technical model of human eyes (or animal eyes), It has the artificial vision of character, graph, image, scene, colour and expression etc. **Artificial Ears,** It is a technical model of human ears (or animal ears), It has the artificial Hearing of voice, noise, sound and language. **Artificial Nose,** It is a technical model of human nose (or animal nose), It has the artificial smelling of various flavor and smell.

By using of FM/AS, we can develop the humanoid (or animaloid) sense device with high performance, such as:

Multi-sense Information Fusion, The humanoid multi-sense information fusion device has the multi-sense abilities, i.e.: Vision, hearing, smell and the information fusion function, It is useful to

improve the effectiveness of feedback measure.

Multi-pattern knowledge Discovery, The multi-pattern knowledge discovery device can discover the knowledge from multi-pattern sense information, Such as image, voice, flavor etc.

5. Control Actuator CA/AO

The concept of Control Actuator based on Artificial Organ (CA/AO) is shown as expression (4)

$$CA + AO \rightarrow CA/AO \qquad (4)$$

Where: CA — Control Actuator, AO — Artificial Organ, CA/AO — Control Actuator based on Artificial Organ, "+" — "Combination", "→"— "Development"

The expression (4) means that based on the combination of methodology and technology of CA and AO, we can develop new control actuator CA/AO.

The Artificial Organ (AO) is the technical model of actuator of human (or animal) body, For example:

Artificial Hands and Arms, The humanoid artificial hands and arms are the simulation, extension and expansion of human hands and arms, it is a humanoid intelligent manipulator.

Artificial Feet and Legs, The humanoid (or animaloid) artificial feet and legs are the simulation, extension and expansion of human (or animal) feet and legs. It is a humanoid (or animaloid) intelligent actuator.

Artificial Heart, The humanoid (or animaloid) artificial heart is the simulation, extension or expansion of human (or animal) heart, It is a bionic intelligent liquid control actuator.

The CA/AO has some excellent control performance, such as:

Coordinative Control, The coordinative control of multi-joints, multi-fingers, two-hands and two-foots can be achieved. it is one of the special excellent performance of human motion control system.

Double-direction Control, The double-direction control is another one of the special excellent performance of human physiological control system, for example, the double-direction control of blood pressure increasing or decreasing.

6. Humanoid Intelligent Control System

As an example of Intelligent control system based on Artificial Life ICS/AL A scheme of the Humanoid Intelligent Control System HICS of a robot is shown as in Fig. 2.

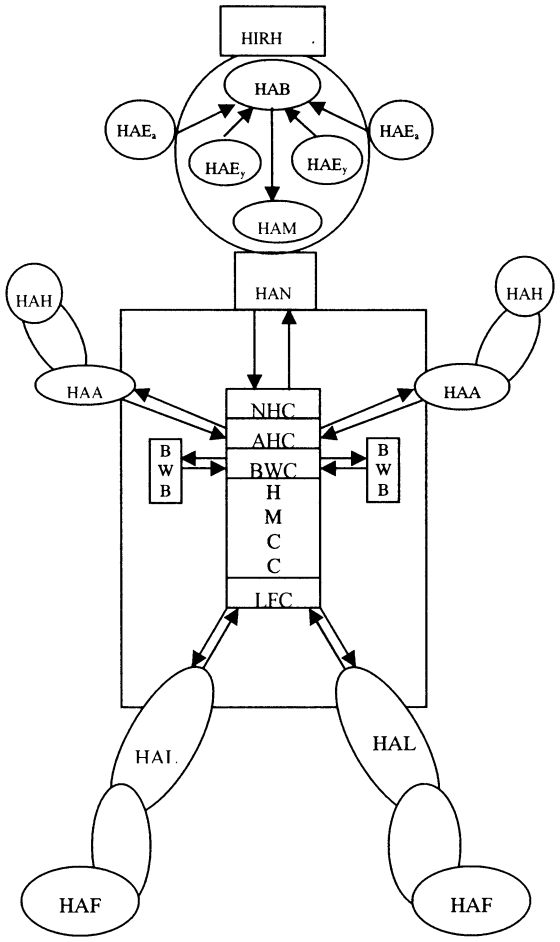


Fig.2 Humanoid Intelligent Robot

In Fig.2:

HIRH Humanoid Artificial Robot Head, **HAB** Humanoid Artificial Brain, **HAE$_y$** –Humanoid Artificial Eyes, **HAE$_a$** Humanoid Artificial Ears, **HAN** Humanoid Artificial Nose, **HAM** Humanoid Artificial Mouth, **HIRB** Humanoid Artificial Robot Body, **HAA** Humanoid Artificial Arms, **HAH** Humanoid Artificial Hands, **HAL** – Humanoid Artificial Legs

HAF – Humanoid Artificial Feet, **HAHAC** – Humanoid Artificial Hands & Arms Controller, **HALFC** – Humanoid Artificial Legs & Feet controller.

Conclusion

The concept of Intelligent Control System based on Artificial Life (ICS/AL) in proposed and the structure of ICS/AL is given.

The concept and principle of Intelligent Controller based on Artificial Brain (IC/AB), Feedback Measure based on Artificial Sense (FM/AS), and Control Actuator based on Artificial Organ (CA/AO) are proposed.

A typical example of ICS/AL: Humanoid Intelligent control system HICS of a robot is given.

References

[1] TuXuyan, "Generalized Artificial Life Race and model", proceedings of the 8-th International Symposium on Artificial Life and Robotics, (AROB 8-th'03) Oita, Japan, 2003.

[2] Xuyan Tu, "AI, AL and Robotics", proceedings of FIRA world congress (plenary speech), Seoul, Korea.

[3] TuXuyan, "Generalized Artificial Life and its Applications", proceedings of the 1-st symposium on "Artificial Life and Applications" of CAAI, 2002.

[4] Xuyan Tu, "Generalized Intelligent System", proceedings of Korea-China Joint workshop on Intelligent Systems, Seoul, Korea, 2002.

[5] Yixin Yin, etc, "Study on Single Neuron Nonlinear PID controller", proceedings of the Korea-China Joint workshop on Intelligent Systems, Seoul, Korea, 2002.

[6] TuXuyan, "Concept, Controls and Methodology of Artificial Life", Proceedings of the conference of Intelligent Automation, 2001.

[7] TuXuyan, "Generalized Artificial Intelligence", Proceedings of the 9-th conference of Chinese Association for Artificial Intelligence (CAAI), 2001.

[8] Xiaoyuan Tu, <Artificial Fish>, Tsing-hua University Publishing House, 2001.

[9] TuXuyan,etc. "Concept, Model, Methodology and strategy of Intelligent Automation", Proceedings of the 7-th conference of Chinese Association(CAAI), 1997.

[10] TuXuyan, <Large Systems Cybernetics>, National Defence Industry Publishing House, 1994.

[11] TuXuyan, "Intelligent Cybernetics", proceedings of the conference on "Computer Vision and Intelligent Control" of Chinese Association for Artificial Intelligence (CAAI), 1991.

[12] TuXuyan, <Artificial Intelligence and its Applications>, Electronic Industry publishing House, 1988.

[13] C.G.Longton,ed. <Artificial Life>, proceedings of an interdisciplinary workshop on the synthesis and simulation of living systems, Los Alamos, 1987.

[14] TuXuyan, "Intelligent Control and Intelligent Management for Large Scale Systems", proceedings of the IFAC/IFZP workshop on AI in Economics and management, Zurich, 1985.

[15] TuXuyan, "A New Approach to Flexible Automation of Large scale systems", proceedings of IFAC-Symposium on system Approach, Vienna, 1983.

[16] TuXuyan,etc. <Bio-cybernetics>, Science publishing House,1980.

[17] TuXuyan,etc. "Intelligent control and its Applications", <Automation>, No.1 1977.

[18] TuXuyan, "Theory of an Harmonically Acting control system with a Large Number of Controlled Variables", proceedings of the 1-st International congress of IFAC, Moscow. 1960.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Evolution of a large-scale spiking neural network for robot navigation.

Malachy Eaton, Niall Jordan, J.J Collins and Mark Mansfield
Dept. of Computer Science and Information Systems
College of Informatics and Electronics,
University of Limerick, Ireland.
Email: Malachy.Eaton@ul.ie,

Keywords: robotics, spiking neural networks, simulated evolution, genetic algorithms

Abstract

This paper describes initial work on the evolution of a large-scale recurrent network of spiking neurons applied to navigating a robot around a simulated environment. The individual neurons in the network are of the spiking variety. They fire once, based on a weighted temporal summation over a number of previous time-steps, each time their threshold is exceeded. After emitting a spike, the neuron cannot fire again for a short time no matter how great the inputs to it are, based on its refractory period.

The robots starting point is randomly selected a small distance from the centre of the environment, if it gets stuck against a wall it is left there until the evaluation cycle is over, or until the rotation of the wheels is sufficient to move it away from the wall. Each individual is currently run for 600 cycles of the network, which on the development machine equates to approximately 10 seconds of running time. Individuals are selected for their ability to move in a straight line whilst avoiding collisions with the walls using an elitist genetic algorithm with single-point crossover operating directly on the spiking neural networks.

Introduction

In the past while some work has been done on the evolution of neural networks for robot navigation, including an entire book on the subject giving accounts of both embodied and simulated experimentation [1]. Also a small amount of work has been done on the evolution of fairly small scale spiking networks for this task , e.g.[2].

In addition to these experiments work continues to be carried out on the construction of relatively large scale spiking neural networks, e.g.[3]. In this paper we describe our initial attempts to apply evolutionary techniques to the automatic construction of a relatively large scale spiking network and its application to a simple task of obstacle avoidance in a simulated environment.

The neural connectivity is based on the fact that neurons in the brain, which are physically close to each other, are generally more likely to be connected than ones further away. Consider a 10,000 neuron neural network. This could be represented as a 10,000 by 10,000 matrix of connections. Consider now a section of this matrix, 100 neurons wide running from one diagonal to the other. All the neurons in this band would have high connectivity towards other neurons in this band assuming neurons which are close together have a relatively high chance of being connected. Neurons from outside this band would still have interactions with the ones inside but on a much smaller scale. This allows for a considerable reduction in the total memory requirement to represent the network.
The connectivity in the network is random. Neurons inside the band have a relatively high probability of being connected to other neurons

(both backward and forward) inside the band. Neurons outside the band have a smaller probability of being connected to neurons inside the band. The current neural network consists of 4,000 neurons, although this should be extendable to 100,000 neurons or more. Each of these is randomly connected to up to 150 close-range neurons and to 5 long-range neurons. This technique reduces the total memory and processing required by the processor and simulates to a certain extent the notion of functional cortical columns in the brain. A neuron then fires based on a temporally weighted summation of the input to that neuron over the previous 10 time steps, recent inputs having a higher weight than inputs in the more distant past. If this value exceeds the neuron's threshold then the neuron fires.

Work is currently in progress on the application of this network to basic navigational tasks on a simulated robot with two wheels and simple sensory input using a basic robot simulator, which we have developed as part of this work.

Simulator and GUI

The robot simulator consists of a white background with the boundaries of the "environment" marked out in black. The present environment is simply a rectangle with sides of approximately 10cm by 15cm. The robot, is represented as a circle 1cm in diameter with a red "nose" identifying the direction in which the robot is facing. The robot has associated with it two wheels. Each of these wheels takes as input a speed value ranging from 0 to 10. When both wheels have the same velocity the robot will travel in a straight line. Varying the speeds of one of the wheels causes the robot to turn. This wheel system is known as a differential drive. It is similar to the way in which a wheelchair works and aims at mimicking the way in which the simple Khepera robots move.

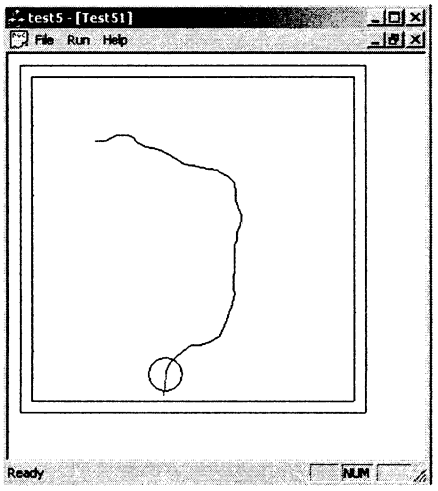


Figure1: The robot simulator

Calculating the Trajectory

The formula used to calculate the path taken by the robot uses basic trigonometry. At any 20ms interval, the change in position of the left wheel is denoted as Δl and of the right wheel as Δr. What needs to be determined is how much it has moved, how much it has turned and in which direction during that time interval. If both Δl and Δr are equal then it has moved forward Δr and hasn't turned at all.
As an example lets assume that Δl is less than Δr and that they're both positive. So the robot will turn to the left and its change in direction is denoted as $\Delta\theta$. $\Delta\theta$ can be calculated using the following formula.

$$\Delta\theta = (\Delta r - \Delta l) / d \qquad (1)$$

where d is the width of the wheelbase of the simulated robot

The distance the center of the robot moved can be calculated as follows:

$$\Delta s = (\Delta r + \Delta l) / 2 \qquad (2)$$

These formulae only work for small wheel movements but are sufficient for determining the approximate position of the robot. Now that the change in direction and the change in position can be calculated it is possible to

integrate the robot motions in order to give its global position. The pose (position and direction) of the robot at any given time can be given by the vector (x,y,θ). If in the the next time interval it moves a distance Δs and changes direction $\Delta\theta$ then its new pose is (x',y',θ'). These can be calculated as follows:

$$x' = x + \cos\theta$$
$$y' = y + \sin\theta$$
$$\theta' = \theta + \Delta\theta \qquad (3)$$

The robot travels around the track at varying speeds and performs reasonably realistic turning manoeuvres. If the robot "hits" any of the walls in the environment then the event is detected and the robot is picked up and transported to a random position in the environment where it can resume its run again.

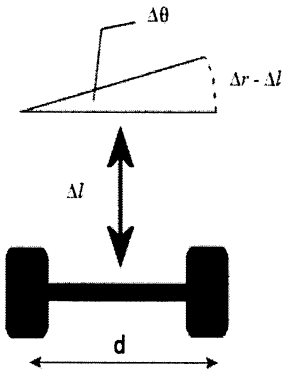


Figure 2: Trajectory calculation

Sensor Data

The robot also has a simulated sensor that points in the direction that it is traveling. The information that this sensor provides is in the form of an approximate distance from the wall it is facing. The maximum distance it can be from a wall is when it is in one corner and is facing the corner diagonally opposite it. This distance is taken and divided into 10 equidistant sections. At any one time interval the robot is a certain number of these sections away from the wall it is facing. This sensor data will be fed into 10 input neurons in the neural network in an inverted fashion.

When the robot is 10 sections away from the wall it is facing then none of these neurons will fire. When it is 9 sections away from the wall one of the neurons will fire, 8 sections away 2 neurons will fire and so on until the robot is only one section away from the wall it is facing. Then all 10 sensory input neurons will fire giving a strong signal into the net to change direction promptly.

Neural Network

The current neural network consists of 4,000 neurons. Each of these is randomly connected to 150 close range neurons and to 5 long-range neurons. There are 30 input neurons firing into the network consisting of the 20 motor neurons, 10 each for the left and right wheel, and 10 neurons for the single sensor. The output layer consists of a similar arrangement. To the measure velocity for each wheel the number of times the ten neurons fire within a 20ms time interval is summed up. The velocities for the 2 robot motors are then updated and the robot will take the new values as inputs the next time it updates its position.

The initial population consists of 12 individuals. Each of these networks is tested on for 600 cycles of the neural network, equivalent to 10 seconds running time. The robot is started from a randomly chosen position. If, by chance, an early un-evolved individual is spawned facing directly into a wall then it has little chance of being able to avoid it. In order to avoid an unfair situation like this each robot is given 3 chances at a fresh start during its 20 second run time.

Network Encoding

Each neuron has associated with it 150 short range connections and weights to other neurons. These connections form the basis for the genetic string that encodes the network. A 1 represents a positive weighted connection and a 0 represents no connection. There are 4000 such strings in each network taking up on average 1 megabyte of storage space. The network is evolved using the following rules:

- After all 12 individuals have been tested the top 6 are be selected to populate the next generation with 2 copies each.
- One point crossover is applied and two offspring are created
- Each individual is then mutated by switching the value of a bit with a probability of 0.05 per bit.

Fitness Function

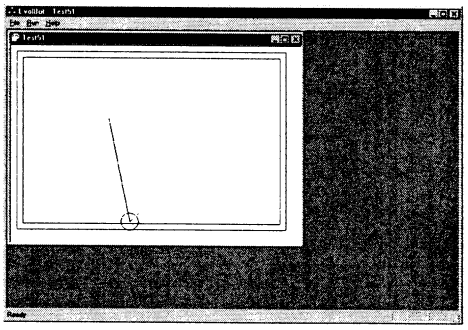
Individuals are selected for their ability to move in a straight line whilst avoiding collisions with the walls. The speed with which the robot progresses is also a factor in the selection process. The formula used is given below:

$$\Phi = V(1- (\Delta V)^{1/2}) (1 - i) \quad (4)$$

where V is the combined velocity of both wheels, ΔV is the difference between the two wheels averaged over all time steps and i is sensory input. The three components of this fitness function are maximum velocity, straightest path and ability to avoid collisions with walls. This is based on the fitness function described in [1] and is designed to reward control algorithms which propel the robot in straight lines at high speeds while avoiding obstacles.

Initial Results

The simulator and neural network has been run a number of times for 12-15 generations of networks. In these initial runs both average and best fitness per generation generally increases in an oscillatory fashion up to generation 6 or 7 and then tends to remain fairly constant. In the initial generations common patterns observed are the robot moving in a relatively straight line before colliding with a wall- this maximises the first term in the fitness equation above, and moving in circles within the boundaries, thus avoiding collision with the walls. These circles increase in diameter with later generations, increasing the second term in the fitness equation. In later generations some rudimentary collision detection and wall avoidance are observed, however these experiments are still at a very early stage.



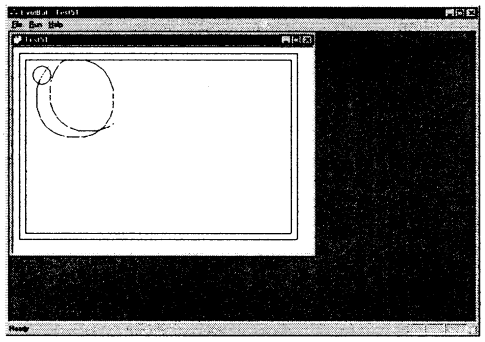


Figure 3: Basic straight line movement and wall collision (below left) and moving in circles (above)

Conclusions and further work

This paper describes initial experiments in the evolution of a relatively large scale (4000 neuron) neural network, and it's application to a simple avoidance problem. While this paper describes work in progress, initial results are encouraging with a wide range of different trajectories being observed through the environment. Further work needs to de done to extend the range of sensory input available to the robot . Currently the robot only 'sees' in one direction, however still appears to perform rudimentary obstacle avoidance. This sensory input needs to be extended to two, or more, directions as in the Khephera robot. Further work also needs to be done to allow for the robot to operate in a wide range of simulated environments to test the learning and generalisation capabilities of the neural network.

References

[1]Nolfi S, Floreano D(2000). Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organising Machines. MIT Press, Cambrige, Massachusetts, 2000.

[2]Floreano D, Mattiussi C(2001). Evolution of Spiking Neural Controllers for Autonomous Vision-Based Robots. In: Gomi T(ed), Evolutionary Robotics 200, Berlin: Springer-Verlag LNCS 2217, 2001, pp. 38-61.

[3] Mattia M, Del Giudice P(2000). Efficient event-driven simulation of large networks of Spiking Neurons and Dynamical Synapses, Neural Computation 12, MIT Press, 2000, pp. 2305-2329.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

High Precision and Predictive Control for Linear Motor Container Transport System using DR-FNNs

Jin Woo Lee
Department of Electrical Engineering
Dong-A University
3205, 840-1, Hadan-dong,
Saha-gu, Busan, Korea
jinwoo@donga.ac.kr

Sang Ki Kim
Department of Electrical Engineering
Dong-A University
3205, 840-1, Hadan-dong,
Saha-gu, Busan, Korea
smiledu@donga.ac.kr

Kwon Soon Lee
Division of Electrical, Electronics & Computer Eng.
Dong-A University
840-1, Hadan-dong,
Saha-gu, Busan, Korea
kslee@daunet.donga.ac.kr

Abstract

In this paper, we will introduce a control strategy that is the multi-step prediction control of linear motor-based container transport system for high precision using dynamically-constructed recurrent fuzzy neural network (DR-FNN). Linear motor-based container transport system (LMCTS) is horizontal transfer system for the yard automation, which has been proposed to take the place of automated guided vehicle in the maritime container terminal. LMCTS is considered as that the system is changed its model suddenly and variously by loading and unloading container. The proposed control system is used two DR-FNNs for multi-step prediction. Consequently, the system has an ability to adapt for external disturbance, cogging force, force ripple, and sudden changes of itself.

1. Introduction

For the port automation, many technologies have been developed until now. As the yard automation technology, one of them, the known well automated guided vehicle (AGV) had been proposed. But AGV system had various problems that included the difficulty of control, complexity, low speed, heavy weight, low position accuracy, etc. Above all, the main problem was the part of navigation that had difficulty to apply in the whole system and to work together with other systems. Recently, LMCTS has been developed to solve these problems. LMCTS is based on the concept of linear motor and rail structure. The main benefits of LMCTS include the high force density, no need for sub-systems, and, most importantly, the high precision and accuracy associated with the simplicity in mechanical structure [1,2].

The system is considered permanent magnet linear synchronous motor (PMLSM) that is consists of stator modules on the rail and shuttle car. Because of large variant of mover's weight by loading and unloading containers, the difference of each characteristic of stator modules, and a stator module's default etc., LMCTS is able to consider as that the system is changed its model suddenly and variously. Then, we will introduce the control strategy that is a multi-step prediction control for LMCPS using two DR-FNNs.

The proposed control system is used the network that is based on fuzzy neural network. It is composed of the structure of neural network that have the fuzzy inference ability and the recurrent loops. In addition, it has ability to adjust the structure of network. Layers in the network are consisted of the adjustable number of nodes which act a role such as the membership function or rule base. We use two units of DR-FNN and each network has two output nodes in the system. One is a predicted system output, another is a predicted reference signal, and others are control input signals from each other network. Then the system has an ability to adapt for the external disturbance, cogging force, force ripple, and sudden changes of itself by unifying network weights of controller and emulator.

2. Modeling of LMCTS

The system consists of a substructure, shuttle cars, and a control system. The motor style is permanent magnet synchronous linear motor as shown by figure 1.

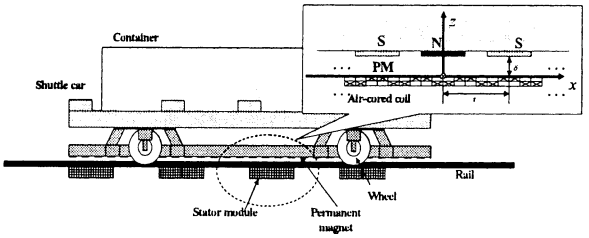


Fig. 1 The configuration of the PMLSM-based LMCTS

Because of the permanent magnets are installed under the plate of shuttle car, no power cables and other devices are required onto the shuttle car. LMCTS is able to start modeling from the general PMLSM[3,4] as follows:

$$\left.\begin{aligned} v_q &= R_s i_q + p\lambda_q + \omega_e \lambda_d \\ v_d &= R_s i_d + p\lambda_d + \omega_e \lambda_q \end{aligned}\right\} \quad (1)$$

Flux linkages for each axis, angular velocities, and electric linear velocity are represented by (2).

$$\begin{aligned} &\lambda_q = L_q i_q, \quad \lambda_d = L_d i_d + \lambda_{PM} \\ &\omega_e = n_p \omega_r, \quad \omega_r = \pi v / \tau \\ &v_e = n_p v = 2\tau f_e \end{aligned} \quad (2)$$

Electromagnetic power is represented by (3).

$$P_e = F_e v_e = 3n_p\{\lambda_d i_q + (L_d - L_q) i_d i_q\}\omega_e / 2 \quad (3)$$

Electromagnetic force is represented as (4).

$$F_e = 3\pi n_p\{\lambda_d i_q + (L_d - L_q) i_d i_q\} / 2\tau \quad (4)$$

Table 1 The used parameters

v_d, v_q	d-q axis voltages
i_d, i_q	d-q axis currents
R_s	phase winding resistance
L_d, L_q	d-q axis inductances
ω_r	angular velocity of the mover
ω_e	electrical angular velocity
λ_{PM}	permanent magnet flux linkage
n_p	number of primary poles
p	Differential operator
v	Linear velocity of the mover
d, x	Distance (mover's position)
τ	pole pitch
v_e	electric linear velocity
f_e	electric frequency
M	total mass of the moving element system
D	Viscous friction and iron-loss coefficient
F_L	external disturbance term

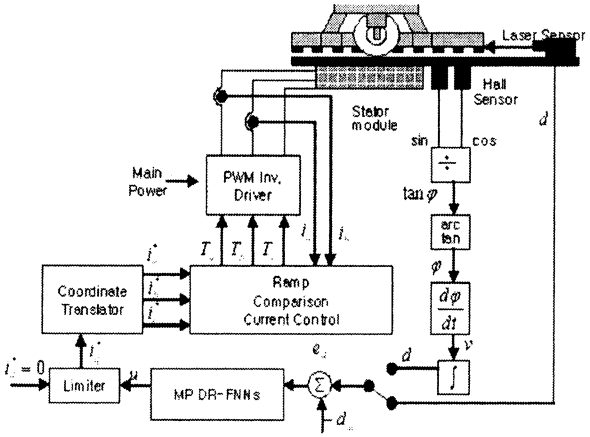


Fig. 2 LMCTS and driver system configuration

Force equation is represented as (5).

$$F_e = 3\pi\lambda_{PM} i_q / 2\tau \tag{5}$$

Simplified PMLSM drive system is able to describe as follows:

$$F_e = K_f i_q^* \tag{6}$$

$$K_f = 3\pi n_p \lambda_{PM} / 2\tau \tag{7}$$

The simplified shuttle car's dynamic equation is represented by (8).

$$F_e = Mpv + Dv + F_L \tag{8}$$

The gain K_f is scale factor between input current and thrust force. But, if the force ripple and the interval of stator module are considered, it can be modified K_f to nonlinear function as (9).

$$K_f(d) = K_{f0} + K_{ripple}(d) \tag{9}$$

$K_{ripple}(d)$ is the function of force ripple effect as shown by (10), which is caused by the condition of install of stator modules, the change of distance, and the winding self-inductance varies[5].

$$K_{ripple}(d) = K_{f0}\{0.03 \cdot \sin(2\pi d / P + \pi / 4) + 0.038 \cdot \sin(6\pi d / P + 0.09\pi)\} \tag{10}$$

F_L is considered as shown by equation (11). Here, f_{dis} is the white noise and the cogging force is described as equation (12).

$$F_L = f_{cogging} + f_{dis} \tag{11}$$

$$f_{cogging} = K_s\{(15\sin(2\pi \cdot d / P + \pi / 4) + (20\sin(6\pi \cdot d / P + 0.09\pi)\} \tag{12}$$

Because of the shuttle car weight change by loading and unloading container and the interval of stator modules, coefficients M and D are needed to consider as the nonlinear function $M(t)$, $D(t)$ respectively. Then, the modified shuttle car's dynamics is described by (13) consequently. The whole system configuration is can be described by figure 2.

$$\ddot{x} = \frac{(K_{f0} + K_{ripple})u(t) - D(t)\dot{x} - f_{cogging} - f_{dis}}{M(t)} \tag{13}$$

3. Control Strategy

3.1. DR-FNN unit

In this paper, the structure of network is composed of having two input and output nodes respectively. In the proposed control system, two networks are used as the controller and emulator. Figure 3 shows DR-FNN that is used the emulator unit.

One of the output nodes is prediction of plant output for one-step ahead, and the other is the predicted control input u. And inputs are recurrent delayed values of control input and system output about p-step time index backward. The network can be considered the recurrent style. Because outputs are used inputs these are delayed in next time index.

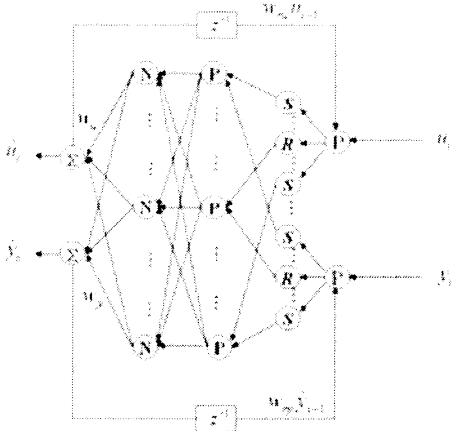


Fig. 3 DR-FNN emulator unit

Layer 1 is input layer for the linguistic variables. Layer 2 is consisted of the membership function that has nodes based on the radial basis function). In case of j=1 and u, for two nodes of edged side in layer 2, the membership function get the shape of sigmoid function, and these are optimized automatically by the back-propagation method. The output of membership function is presented by equation (14) in the second layer [7, 8].

$$\mu_{ij}(x_i) = \exp\left[-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right] \tag{14}$$

(i= 1, 2, ,r, j= 2, ,u-1)

Where, x_i is the product between the current input and delayed output. x_1 and x_2 are presented by equation (15) and (16) respectively.

$$x_1(t) = u(t) \cdot w_{ru} \cdot \hat{u}(t-1) \tag{15}$$

$$x_2(t) = y(t) \cdot w_{ru} \cdot \hat{y}(t-1) \tag{16}$$

Third layer is IF-part for fuzzy rules. For the j-th rule R_j, its output is described by equation (17) in third layer.

$$R_k = \exp\left[-\frac{\sum_{i=1}^{2p} (x_i - c_{ij})^2}{\sum_{i=1}^{2p} \sigma_{ij}^2} \right] \tag{17}$$

The normalized output of third layer was calculated in layer 4. It can be expressed by equation (18).

$$N_k = \frac{R_k}{\sum_{l=1}^{N} R_l} \tag{18}$$

The fifth (output) layer computes outputs of consequence part. Control input and predicted plant output are calculated by (19) and (20) respectively.

$$\hat{u}(t) = \sum_{k=1}^{N} w_{1,k} \cdot N_k \tag{19}$$

$$\hat{y}(t) = \sum_{k=1}^{N} w_{2,k} \cdot N_k \tag{20}$$

3.2. Predictive Control Scheme

For one-step prediction, DR-FNN is constructed beside plant as an emulator style. The proposed control system with two DFNNs is shown by figure 4.

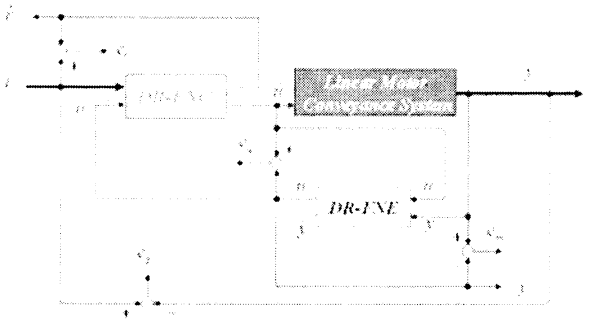


Fig. 4 The control system with DFNN controller and emulator

It performs not only controller but also act of on-line identifier by two outputs from network. The system can obtain predictions of reference input signal $\hat{y}_d$ and the other control input $\hat{u}$. Then, system has two control inputs protruded from controller and emulator. In figure 4, the system structure has four errors from networks and plant. Errors and network learning is can be described as follows:

$$\left.\begin{aligned} e_u &= u_c(t-1) - u_e(t-1), \quad E_u = \frac{1}{2} e_u^2 \\ w_{l,k}(t+1) &= w_{l,k}(t) + \eta\left(-\frac{\partial E_u}{\partial w_{l,k}} \right) \end{aligned}\right\} \tag{21}$$

$$\left.\begin{aligned} e_p &= y_c(t) - y(t), \quad E_p = \frac{1}{2} e_p^2 \\ w'_{l,k}(t+1) &= w'_{l,k}(t) + \eta\left(-\frac{\partial E_p}{\partial w'_{l,k}} \right) \end{aligned}\right\} \tag{22}$$

$$\left.\begin{aligned} e_r &= r(t+1) - \hat{r}(t+1), \quad E_r = \frac{1}{2} e_r^2 \\ w_{l,k}(t+1) &= w_{l,k}(t) + \eta\left(-\frac{\partial E_r}{\partial w_{l,k}} \right) \end{aligned}\right\} \tag{23}$$

$$\left.\begin{aligned} e_m &= y(t) - \hat{y}(t-1), \quad E_m = \frac{1}{2} e_m^2 \\ w'_{l,k}(t+1) &= w'_{l,k}(t) + \eta\left(\frac{\partial E_m}{\partial w'_{l,k}} \right) \end{aligned}\right\} \tag{24}$$

The predicted reference input signal $\hat{y}_d$ and the predicted plant output $\hat{y}$ are trained by e_{yd} and e_d respectively. To improve the adaptability in this case, the proposed control scheme is regarded as the predicted next step system input because two units of DR-FNN are used as a controller and emulator as shown by figure 5.

In the part of controller, if DR-FNC were trained enough by the various patterns of reference input signals, network can simulate without the plant virtually using predicted next reference input. Even though network was trained well for some known patterns also, input is the value that can not be guarantee absolutely. And the reference input can be decided by unknown pattern.

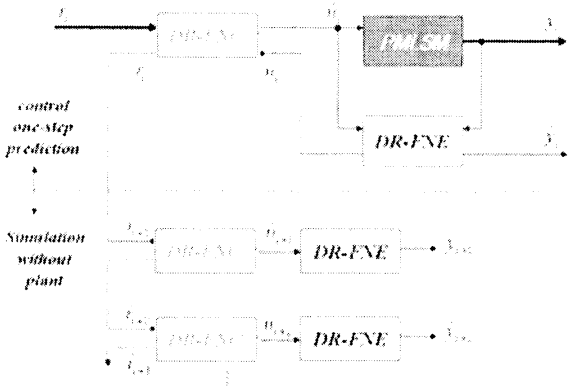


Fig. 5 Multi-step ahead prediction without plant.

To decide the number of nodes, an objective function is needed as (25). The preconfigured threshold values $\Delta Q_{\min}$ and $\Delta Q_{\max}$ are needed too. In case of (26), the nodes of network are added and the prediction step is reduced. In case of (27), the network keeps its structure. The case of (28) is the number of network nodes is reduced and the prediction step must be increased.

$$Q = \sqrt{\alpha E_u^2 + \beta E_p^2 + \gamma E_r^2 + \delta E_m^2} \tag{25}$$

- Case 1: increase node and reduce prediction step

$$\frac{\partial Q(t)}{\partial t} \geq \Delta Q_{\max} \tag{26}$$

- Case 2: keep the structure of network

$$\Delta Q_{\min} \le \frac{\partial Q(t)}{\partial t} < \Delta Q_{\max} \tag{27}$$

▪ Case 3: decrease node and extend prediction step

$$\frac{\partial Q(t)}{\partial t} < \Delta Q_{\min} \approx 0 \tag{28}$$

4. Simulation and Results

In the simulation, the scenario of simulation is as follow: At first, the empty shuttle car goes until the point of 33m and comes again. Secondly, the container of 40ton is loaded at 50sec. Finally, the loaded shuttle car returns until 55m.

The case of figure 6(a) is the description of response for the distance of shuttle car using fixed parameters and control the parameter at learned FNN in advance, (b) is result of by using the FNN of recurrent type and regulation node about the position and the velocity respectively. By same method, (c) is result of separating to 16 node start from 5 membership functions. According as go to (c) from (a), it is improved the response characteristics and reinforced steep weight change of the system and effect of cogging force and stator module. The dotted line is a reference and a solid line is a change of the distance.

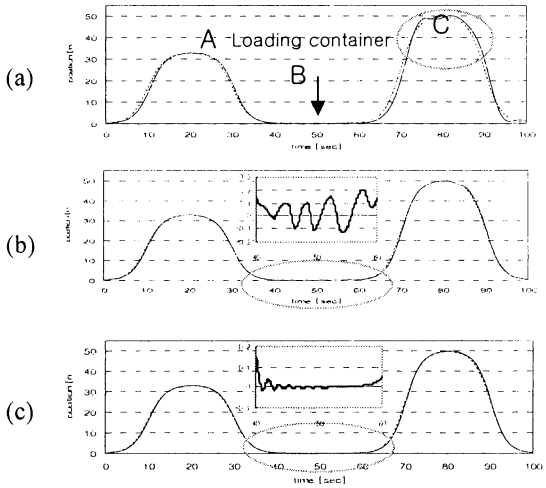


Fig. 6 Reference trajectory and responses of shuttle car:
(a) the variation of displacement using FNN of the fixed structure;
(b) R-FNN of the fixed structure; (c) DR-FNN.

Figure 7 and 8 show the change of the control input and distance error. The maximum position error is less than 1.8cm and the average is about 0.8cm in the stop area.

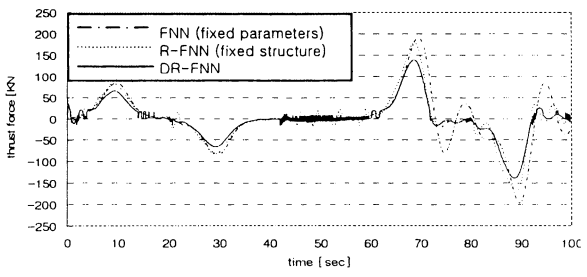


Fig. 7 Variations of thrust force

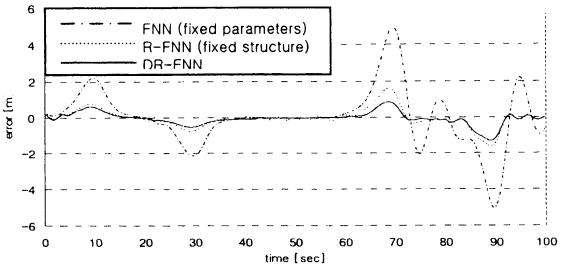


Fig. 8 Results of the distance error

5. Conclusions

In this study, we modeled LMCTS system by modifying the general PMLSM and adding with the considerable various disturbances. In LMCTS, the system included various kinds of problems to control such as cogging force, force ripple, external disturbances, the variation of mover weight, the periodic lack of thrust force, etc. To improve the positioning accuracy and reduce the energy consumption, we proposed a control system that has multi-step predictable structure using two DR-FNN units. In case of using DR-FNN, there are improvements of 68%, 3% than supervised FNN, R-FNN respectively in position accuracy. By multi-step predictable structure, the amount of control input (thrust force) was reduced about 13%, 8% than FNN, R-FNN respectively.

Acknowledgements

This work was supported by the program for cultivating graduate students in regional strategic industry of Ministry of Commerce, Industry and Energy (MOCIE) in Korea.

References

[1] Klaus-Peter Franke, "Boosting Efficiency of Split Marine Container Terminals by Innovative Technology," *IEEE Intelligent Transportation Systems Conference Proc.*, Oakland, USA, August, pp. 774-779, 2001.

[2] K. K. Tan, S. N. Huang, and T. H. Lee, "Robust Adaptive Numerical Compensation for Friction and Force Ripple in Permanent-Magnet Linear Motors," *IEEE Trans. on Magnetics*, Vol. 38, No. 1, pp. 221-228, 2002.

[3] F. J. Lin, R. J. Wai, and C. M. Hong, "Hybrid Supervisory Control Using Recurrent Fuzzy Neural Network for Tracking Periodic Inputs," *IEEE Trans. on Neural Networks*, Vol. 12, No. 1, January, 2002.

[4] Rong-Jong Wai, Faa-Jeng Lin, "Adaptive Recurrent-Neural-Network Control for Linear Induction Motor," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 37, No. 4, pp. 1176-1192, 2001.

[5] Bin Yao & Li Xu, "Adaptive robust motion control of linear motors for precision manufacturing," *Mechatronics*, Vol. 12, pp. 595-616, 2002.

[6] Y. Frayman and L. Wang, "A Dynamically-constructed Fuzzy Neural Controller for Direct Model Reference Adaptive Control of Multi-input-multi-output Nonlinear," *Soft Computing*, Vol. 6, pp. 244-253, 2002.

[7] T. L. Seng, M. B. Khalid, and R. Y. Yusof, "Tuning of a Neuro-Fuzzy Conroller by Genetic Algorithm," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 29, No. 2, pp. 226-236, 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Evocation of Robot Behaviors Based on the Affordance of Worksite Objects

Kazuo Tani and Takuya Kawamura
Department of Human and Information Systems, Gifu University
Yanagido, Gifu 501-1193, Japan tani@info.gifu-u.ac.jp

Hirotaka Yamada
Toyota Industries Corporation
Kariya, Aichi 448-8671, Japan

Abstract

The concept of affordance is introduced to generate robot behaviors. We consider such robots that are born disassemblers and accept affordances from parts of mechanical products, i.e., evoke appropriate behaviors for removing them. The robots evolve themselves for higher efficiency by developing diachronical and synchronical regulations for accepting affordances, which can also be regarded as newly found affordances. This idea is demonstrated through simulations using the genetic algorithm.

Key Words: Affordance, Autonomous robot, Behavior, Disassembly, Genetic algorithm

1 Introduction

It is one of the greatest subjects of human's research to develop autonomous robots that can plan and execute their behaviors in order to adapt to their environments. Many approaches have been proposed and studied for the control schemes of robots, such as model-based approach [1] and behavior-based approach [2]. These approaches have been devised from the knowledge obtained by observing and analyzing the human and animal conducts.

Recently, the concept of affordance proposed by J.J. Gibson and others in the field of ecological psychology [3][4] is noted. According to the concept, behaviors are evoked in such a fashion that the environment provides the animals with the opportunity of performing the behaviors and the animals takes the opportunity, rather than the animals execute their behaviors arbitrarily. And the affordance denotes the opportunity. Furthermore, such affordances are found during the interaction of the animals and the environment in the process of the animals' evolution in the environment.

There are some researches that use the concept of affordance. Tagawa et al. [5] introduce the concept of affordance to the autonomous control of a robot. They assume a robot to be a finite state machine and claim that to learn the transition laws is to find affordances. Lee et al. [6] use Q-learning to obtain affordances in the form of probabilistic relation between sensing and behaviors.

We have been conceiving the automatic system for disassembly of mechanical products by a robot based on the scheme of behavior-based robotics [7][8]. We have come up with the idea that disassembly is a good example to apply the concept of affordance to. Each part of the product is considered to afford the behavior for removing itself [9]. Disassembly proceeds as the robot accepts the affordances of parts that constitute the product. This process of disassembly is a probabilistic process, unlike [5], and also tolerates the occasional failures in removing parts. However, the fewer failures the better, for it mean the higher efficiency in disassembly. In this paper, we will show that some regulations for accepting affordances serve for higher efficiency, and that the regulations themselves are a new kind of affordances.

The animals' acquiring behaviors of capturing food is understood as finding the affordances for those behaviors in the process of animals' evolution in the environment rich with such food based on the situation that the behaviors of capturing the food serve for the preservation of the animal species. If we transfer this process from animals to robots and set the situation that disassembly serves the preservation of robot species, we can expect that robots will evolve themselves to acquire the behaviors of disassembly, or in other words, that the affordances for disassembly will be found.

2 Disassembly based on affordances

A child plays with a toy and ends up with it taken apart. An adult does not know the inner structure of a product but finds disassembly actions only by looking at it and often succeeds in completely disassembling it only by doing the actions. From these facts, we can consider that each part constituting a product affords the action to remove it. "A hexagon affords the action of turning it counterclockwise" can be translated in the engineering expression into "A hexagon bolt or nut can be removed by turning it counterclockwise."

We consider a robot that accepts affordances and execute disassembly behaviors. In the simulation we assume its behaviors as follows. These are classified into robot capabilities (Class A) and behavior regulations (Class B).

A1: Robot finds a product.
A2: Robot finds a part in the product.
A3: Robot accepts the affordance of the part it finds, i.e., it knows the removing action for the part and executes it unless it is inhibited by the regulation B1 below.

Note that the removing action may succeed or fail. The failure is due to the constraint by another part.

A4: Robot can know the success and failure of the removing action in a finite time.
A5: Robot can remember the failed part and identify it when it finds it next time.
B1: Robot does not try disassembly action on the once failed part unless it has succeeded in removing any other part after the failure.

Disassembly of more complicated products requires attacks upon many sides. Therefore the following behaviors are added.

A6: Robot turns over the product to change the attack side.

B2: During the attacks upon one side, if the number of non-trials (by B1) exceeds the limit value r, robot turns over the product to change the attack side.

B3: During the attacks upon one side, if the number of part removal failures exceeds the limit value p, robot turns over the product to change the attack side.

B4: If the number of turning over exceeds the limit value f, robot quits the disassembly task uncompleted.

Note that the counts of numbers in B2, B3, and B4 are reset when any one part is removed.

We made simulation for an extension socket shown in Fig. 1 (10 parts, 2 attack sides) and a grass cutter shown in Fig. 2 (27 parts, 5 attack sides). See Appendix A for the disassembly simulation.

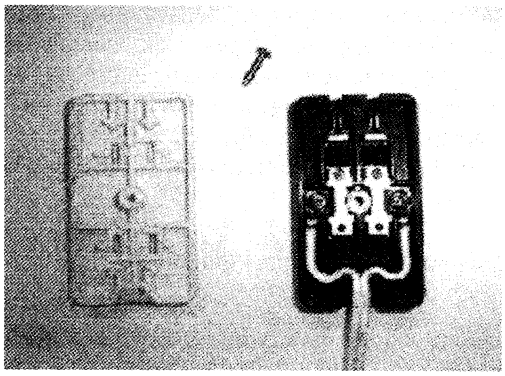


Fig. 1 Extension socket

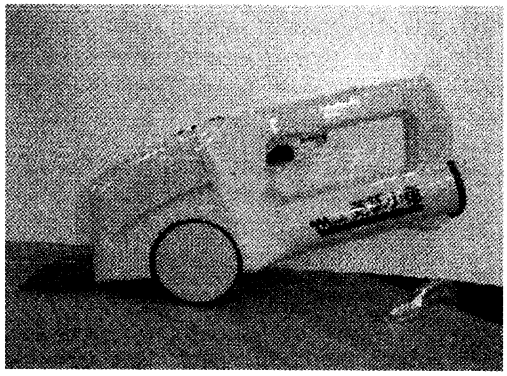


Fig. 2 Grass cutter

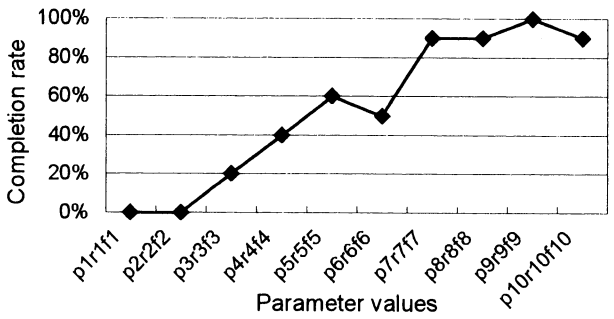


Fig. 3 Disassembly completion rate for the grass cutter

Note that the simulation is for the interactions between the robot and the environment, where the robot is an unthinking agent without any prior information (model) about the product. In the simulation we assume that the robot finds any part appearing in its view randomly with the same probability. The probability of complete disassembly (completion rate) depends on the values of p, r, and f. The simulation results for the grass cutter with the same value for p, r, and f are shown in Fig. 3, in which greater values for p, r, and f give higher completion rates. For $p = r = f = 5$, a typical completion requires 7 tried sides, 114 part findings, and 41 failed removals, while for $p = r = f = 10$, a typical completion requires 11 tried sides, 202 part findings, and 55 failed removals. For the extension socket $p = r = f = 10$ gives a completion rate of 90 % or higher.

3 Diachronical regulations for accepting affordances

The limit values p, r, and f described in the previous chapter regulate the acceptances of affordances depending on how the disassembly proceeds. They serve for higher efficiency in disassembly by letting the robot change the attack sides and sometimes quit deadlocks in case of difficult or impossible products.

Consider robots whose a priori purpose is disassembly. They find affordances in the process of evolution. If the robots complete disassembly of a product, they get merit although at the cost of the paid efforts. Failure in complete disassembly means all loss and no gain. Giving up is important. This resembles the case of a predator chasing a prey. He should not continue chasing a prey he is not likely to catch.

Here, considering the disassembly task and introducing the cost for incompletion instead of the merit for completion, we define the cost the robot pays as follows.

Cost for finding a part = 1
Cost for trying removal of a part = 3
Cost for changing the attack side = 5
Cost for complete disassembly $w = 0$ or incomplete disassembly $w = 100$
Cost for the limit values = $p + r + f$.

The cost for the limit values is introduced to keep them reasonably small. With d parts found, t parts tried, and s sides attacked, the total disassembly cost will be as follows.

$$\text{Disassembly cost} = 1d + 3t + 5s + w + p + r + f \quad (1)$$

Consider a group of robots that evolve themselves so that the disassembly cost is smaller. This process is simulated using the genetic algorithm (GA). Here the fitness is defined as the inverse value of the disassembly cost.

$$\text{Fitness} = 1/(1d + 3t + 5s + w + p + r + f) \quad (2)$$

Every robot is given a product, it behaves, and then GA treats it for the next generation. See Appendix B for the GA process. The simulation results for the grass cutter are shown in Fig. 4. Note that, since the disassembly process is a probabilistic process, the optimal limit values should give the maximum expectation values of fitness. However, they can occasionally yield smaller fitness values for individual

disassembly processes. In GA simulation, a higher fitness for an individual robot results from the mixture of its good features and its luck, and results in the selection of the robot. That the group average value of the fitness gets greater as the generations proceed, as shown in Fig. 5, means that the group acquires good features as the generations proceed. The limit values for maximum fitness in the final generation are $p = 13$, $r = 5$, and $f = 47$.

That the robots adapt the limit values means that, since the action to remove each part is the acceptance of the affordance, the robots find the behavior to regulate the acceptance of affordances diachronically, or in other words, the robots find new affordances from the environment filled with specific products.

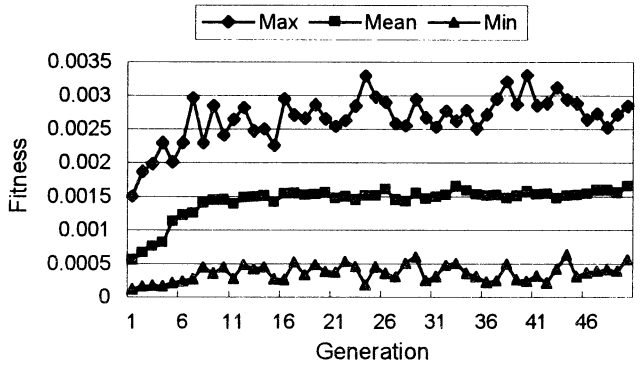


Fig. 4 Process of optimizing parameters for the grass cutter

4 Synchronical regulation for accepting affordances

So far we have made simulation assuming that robots accept an affordance with the same probability from every visible part. However, we can suppose that the probability of accepting the affordance is variable for each part. It means that the robots accept the affordances of some parts with greater probability than those of other parts.

Here we define the value of noticeability for each part of the product. The probability for the robot to find a particular part is given as the ratio of the noticeability value of that part to the sum of the noticeability values of all the parts seen in the view of the robot.

Now we optimize the noticeability of each part of the extension socket using GA. The fitness is defined by eq. (2). The mean values of the noticeability for the final generation are shown in Table 1 and the evolution process is shown in Fig. 5. We see that the average fitness for the group increases as the generations proceed. The disassembly process is programmed so that all the parts in the view are listed and the decision about which part the robot finds is made using the roulette selection method based on the noticeability. A human makes such selection but not consciously. A human baby finds a human face or a pair of eyes quickly in the scene [10]. The human visual processing system has acquired such characteristics through the evolution. The establishment of noticeability values means that the regulation for accepting an affordance among those presented simultaneously is found as a new affordance from the environment.

Table 1 Acquired noticeability of parts for the extension socket

No.	Part	Noticeability
1	upper case	16
2	lower case	2
3	cord	12
4	large screw	20
5	contact metal 1	3
6	contact metal 2	3
7	medium screw 1	17
8	medium screw 2	9
9	small screw1	8
10	small screw1	9

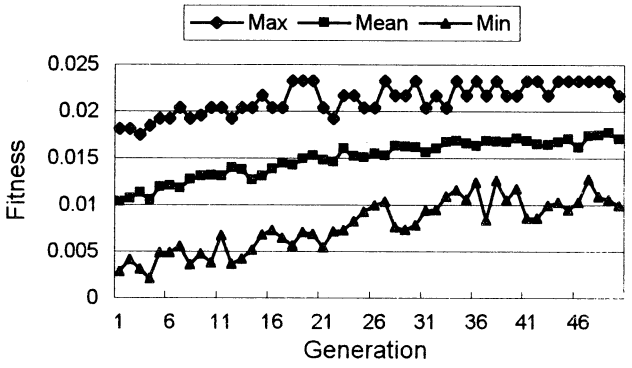


Fig. 5 Process of optimizing noticeability of parts for the extension socket

5 Conclusions

We have shown that a group of robots whose objectives is disassembly of products execute disassembly actions by accepting affordances from the parts, and that, if they are in the environment filled with specific products, during the process of evolution they acquire the behaviors to regulate the acceptance of affordances diachronically and synchronically. It is just like animals that evoke behaviors by accepting affordances from the environment and acquire the mechanism to attain their objectives. We also claim that the regulations are new kind of affordances found in the environment.

The hypothesis of affordance may not have acquired many supports yet. This paper, however, tries to be a support for the hypothesis by showing the possibility of the hypothesis to be applied to robot control.

The practical intension of our research is the facilitation of robot control for task execution by making use of the interaction evoked between the task object and task machines. Recently, as the global environmental problems grow, recycling of disused products is getting more and more important. Since April 2002 the home appliance recycling act is effective in Japan and recycling is promoted legally. By taking disused products apart into parts and selecting them, higher quality of recycled materials and saving of energy for refining recycled materials will be attained. Robots assumed in this paper to evoke disassembly behaviors based on affordances have the

possibility towards an automatic disassembly system because they may not need detailed data of products. We hope to go on to the experiment using real machines.

Appendices

A Simulation of disassembly process

The simulation for a robot to perform disassembly is as follows. A constraint matrix and occlusion matrices are used. The constraint matrix $\{c_{ij}\}$ has $c_{ij} = 1$ if part j constrains part i from being removable and $c_{ij} = 0$ otherwise. An occlusion matrix $\{h_{ij}\}$ has $h_{ij} = 1$ if part j hides part i invisible and $h_{ij} = 0$ otherwise. A product has so many occlusion matrices as it has attack sides. The flow of the simulation is as follows.

L1: Register all the parts in the remaining parts list.
L2: Make the visible parts list by listing up all the parts seen by the robot from the remaining parts list, the present attack side, and the occlusion matrix of that attack side. If there is no part, finish the disassembly complete.
L3: Select a part from the visible parts list at random with equal probability as one found by the robot. If the noticeability is considered as in Chapter 4, however, select a part at random with the probability proportional to the noticeability value of the part as one found by the robot. Apply the regulation B1 to the part, and if it still remains once failed, skip to L5.
L4: Try removal action on the found part. That is, the constraint matrix is looked up and, if the found part is not constrainod by any other part the robot succeeds in removing it, and otherwise it fails. In case of success remove the part from the remaining parts list, reset once failed parts, reset all the limit number counts, and go to L2. In case of failure go to L6.
L5: Apply the regulation B2. If true go to L7, if false go to L2.
L6: Apply the regulation B3. If true go to L7, if false go to L2.
L7: Apply the regulation B4. If false move to another attack side at random and go to L2. If true quit disassembly incomplete.

Note that the robot does not know anything about the constraint or occlusion matrices or the parts lists for itself. They are only introduced to simulate how the environment looks to the robot. The robot only behaves by looking at the environment.

B Genetic algorithm

In the use of the GA, the number of the robots is 100.

(1) In the optimization of the diachronical regulations, the limit values p, r, and f are each expressed in an 8-bit byte, and 24-bit genes consisting of these three bytes are considered. The disassembly simulation is performed for each robot and the fitness is evaluated. In the progress of generations, the operations of selection, crossover, and mutation are performed. Selection is performed using the roulette selection method based on the fitness. Crossover is performed such that one-place crossover occurs bitwise over the 24 bit genes with the crossover rate of 0.9. Mutation is performed such that one bit in the genes is switched with the mutation rate of 0.01. The process is continued over 50 generations.

(2) In the optimization of the synchronical regulations, the limit values p, r, and f are fixed (for the extension socket $p = 16$, $r = 2$, and $f = 6$, and for the grass cutter $p = 13$, $r = 5$, and $f = 47$). The number n denotes the number of the parts or the kinds of parts. Consider genes each consisting of a string of n integers x_i ($1 \leqq i \leqq n$), where x_i is the noticeability of the part or of the kind of parts with the restrictions of $1 \leqq x_i \leqq 20$ and $\Sigma x_i = 10n$. For each robot the disassembly simulation is performed and the fitness is evaluated. In the progress of generation, the operations of selection, crossover, and mutation are performed. Selection is performed using the roulette selection method based on the fitness. Crossover is performed such that one-place crossover occurs with the unit of integer over the n-integer genes with the crossover rate of 0.9 and then normalization is performed to meet the restrictions. Mutation is performed such that a pair of integers in the genes are chosen at random with the mutation rate of 0.01 and one of them is incremented and the other decremented. The process is continued over 50 generations.

References

[1] Y. Shirai, H. Inoue: "Research on intelligent robots—Towards model based robotics," J. Robotics Society of Japan, vol.5, no.6, pp.462-469 (1987) (in Japanese)
[2] R.A. Brooks: "A robust layered control system for a mobile robot," IEEE J. Robotics and Automation, vol.2, no.1, pp.14-28 (1986)
[3] J.J. Gibson: The Ecological Approach to Visual Perception, Houghton Mifflin Company (1979)
[4] E.S. Reed: Encountering the World: Toward an Ecological Psychology, Oxford University Press (1996)
[5] K. Tagawa, S. Kawaguchi, K. Inoue, H. Haneda: "Emergence of intelligent robot by using genetic algorithm and affordance," J. Robotics Society of Japan, vol.17, no.7, pp.1023-1030 (1999) (in Japanese)
[6] M. Lee, M. Kumon, N. Adachi; "Utilization of affordance by reinforcement learning robot," Trans. Japanese Society for Artificial Intelligence, vol.16, no.1, pp.94-101 (2001) (in Japanese)
[7] K. Tani: "Some considerations on a robotic disassembly system for disused products —In search of the science of disassembly—," Proc. Japan-USA Symposium on Flexible Automation, pp.815-818 (1996)
[8] K. Tani, E. Güner: "Concept of an autonomous disassembly system using behavior-based robotics," Advanced Robotics, vol.11, no.2, pp.187-198 (1997)
[9] K. Tani, T. Kawamura, H. Yamada, S. Shamoto: "Evocation of behaviors for disassembly by the affordance of mechanical products," 6th International Conference on Virtual Systems and Multimedia, pp.585-590 (2000)
[10] R.L. Frantz: "The origins of form perception," Readings from Scientific American: Physiological Psychology, pp.71-77, W.H. Freeman & Co. (1961)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Short-term Memory Circuit Using Hardware Ring Neural Networks

Naoya Sasano
Graduate School of Science and Technology
Nihon University
1-8-14, Kandasurugadai, Chiyoda-ku,
Tokyo, 101-8308 Japan
nsasano@hippo.ecs.cst.nihon-u.ac.jp

Katsutoshi Saeki and Yoshifumi Sekine
College of Science and Technology
Nihon University
7-24-1, Narashinodai, Funabashi-shi,
Chiba, 274-8501 Japan
{ksaeki,ysekine}@ecs.cst.nihon-u.ac.jp

Abstract

A number of studies has recently been made on various neuron models and neural networks. This research is studied for application to engineering problems and the understanding of information processing functions of living organisms. We are studying an asynchronous neural network using a pulse-type hardware neuron model (P-HNM). Recently, we have been trying to construct a short-term memory circuit using hardware ring neural networks (RNN) with P-HNM. In this paper, we discuss the construction of a short-term memory circuit using the hardware RNN, and conduct experiments that explain the characteristics of the network through circuit simulation using PSpice. As a result, we verify that the RNN, which is proposed in this paper, can be used as the short-term memory circuit.

keywords - Pulse-type Hardware Neuron Model, Ring Neural Network, Short-term Memory Circuit

1 Introduction

The information processing functions of living organisms such as memory and learning are being solved gradually by progress in neuroscience. Specifically, from physiological experiments, it is known that stimulus information input affects the duration of neuron firing.

On the other hand, a number of studies has recently been made on various neuron models and neural networks. This research is studied for application to engineering problems and the understanding of information processing functions of living organisms. The present neural networks are applied in various uses, but it is difficult to verify if they are actually reproducing the function of neural networks of living organisms. It is especially important to use biological data for neuron models and neural networks. In this research, the short-term memory model using a recurrent network is proposed and focuses attention on points where the dynamics are analogous to neural networks of living organisms[1,2].

We are studying an asynchronous neural network using a pulse-type hardware neuron model (P-HNM)[3]. The P-HNM has an absolute refractory period and a relative refractory period. The absolute refractory period is the period, where no pulse is output for any input after the P-HNM outputs a pulse. The relative refractory period is the period until threshold returns to the stationary state, after the absolute refractory period ends. Recently, we have been trying to construct a short-term memory circuit using hardware ring neural networks (RNN) with P-HNM[4].

In this paper, we discuss the construction of a short-term memory circuit using the hardware RNN, and conduct experiments that explain the characteristics of the network through circuit simulation using PSpice.

2 A short-term memory circuit

Figure 1 shows an example of an asynchronous pulse train schematic waveform. This figure shows that a pulse train, which isn't an equal spacing in time series, appears repeatedly in fixed periods of time.

We think that information is included in the pulse train, which is input to the RNN. The pulse train on the external input and the pulse train in the circuit are integrated into the synaptic parts. If these two pulse trains are in agreement, we consider this to be a short-term memory circuit, which outputs the information signal. When researching the short-term memory circuit, we considered two types of models. One is the circuit model, in which an output pulse train

occurs, when an external input pattern is completely in agreement with the pattern of the memory circuit of the RNN. The other is the circuit model, in which an output pulse train occurs, when the memory pattern is included in the external input. In this study, we discuss the latter case.

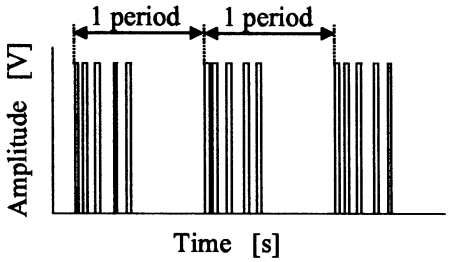


Figure 1: An asynchronous pulse train schematic waveform.

Figure 2 shows the schematic diagram of a short-term memory circuit, which is proposed in this paper. This figure shows a RNN, which is serially connected by P-HNMs ($N_1 \sim N_n$) of n pieces. External input pulses are added into P-HNMs of $m(1 \leq m \leq n)$ pieces for the pulse train to memorize. The output of the RNN is the output of N_1. The delay time of i-th P-HNM N_i is set to $\Delta\tau_i$, and the relative refractory period is set to γ_i. At this point, in the case of $m = 1$, an external input pulse is added into N_1. The remaining P-HNM external input pulses are added on to each P-HNM for the pulse train to memorize. The number of construction to the P-HNM, which counts from N_1, adding an external input to the j-th ($j = 2, 3 \ldots m$), express to $p(j)$ ($p(j) < n$).

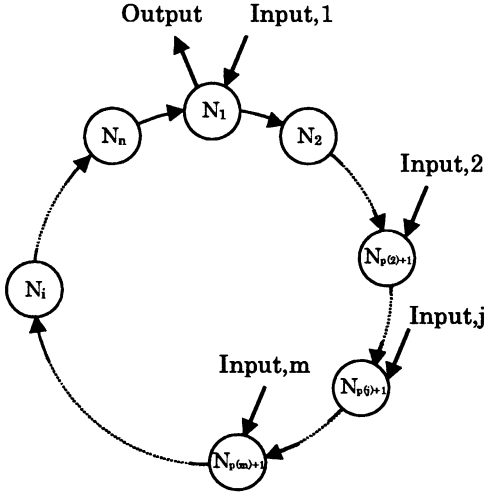


Figure 2: A short-term memory circuit.

If a pulse is input into N_1 and the pulse propagates one by one with $N_1 \rightarrow N_2 \rightarrow N_3 \ldots$, the delay time, which is generated by the whole RNN, is set to the natural period τ_0 and $max[\gamma_i]$ is set to γ. In this case, the necessary condition, on which a pulse repeats propagation inside of the RNN, is described as follows:

$$\tau_0 = \sum_{i=1}^{n} \Delta\tau_i \geq \gamma \tag{1}$$

Moreover, the delay time caused while a pulse propagates to P-HNM, while adding an external input to the j-th, is set to $\tau_{p(j)}$. In this case, $\tau_{p(j)}$ is described as follows:

$$\tau_{p(j)} = \sum_{i=1}^{p(j)} \Delta\tau_i \tag{2}$$

Now, we consider the RNN using the P-HNM, which the ratio of input and output shows 1:1 responses. In this case, if a pulse input by means of the external input only into N_1 and Eq. (1) are satisfied, a pulse repeats transmission for the inside of the RNN. If Eq. (1) is not satisfied, a pulse which transmited to N_i, halts transmission by N_i because N_i is the absolute refractory period or the relative refractory period.

Now, N_1 uses a P-HNM with an adaptive refractory period, the N_1 generates a pulse by the first pulse of pulse trains. Thereafter, a pulse transforms the N_1 into a relative refractory period, which generates a pulse, when an external input pulse and an output pulse of the N_n are added simultaneously. Other P-HNM, which adds an external input, uses a P-HNM, to which a threshold is adjusted so that one pulse might be output by two pulses. The conditions with which an output of the RNN can be obtained, are as follows : (a) an input cycle period T_{in} conforms with τ_0, (b) each of the delay time $\Delta T_{in,j}$, which a pulse train caused from first pulse to j-th pulse, conform with each $\tau_{p(j)}$, and (c) the RNN satisfies Eq. (1). Therefore, the RNN which is proposed in this paper can be extracted from external input pulses. Namely, the RNN becomes a short-term memory circuit, which outputs a cycle period pulse when synchronized to T_{in}. Inparticular, in the case of $m = 1$, the RNN becomes a short-term memory circuit, which outputs a cycle period pulse, when synchronized to T_{in}, if T_{in} of a pulse train, which inputs from an external input, conforms with τ_0.

In this case, we considered the operating conditions of the RNN.

Figure 3 shows the input-output schematic waveform of the RNN. This figure shows that the pulse train, which the RNN memorizes, is input into the

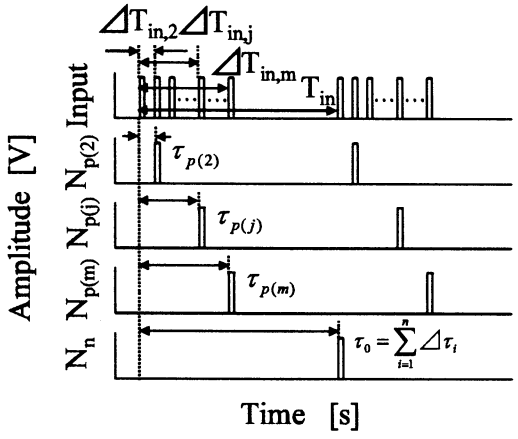


Figure 3: A schematic waveform of RNN.

RNN from an external input. The RNN may generate pulse, if T_{in} conforms with τ_0 and all of $\Delta T_{in,j}$ conform with each $\tau_{p(j)}$. In this case, the necessary condition of $\Delta T_{in,j}$, on which the RNN operates, is described as follows:

$$1 - \frac{\alpha w_p}{\tau_{p(j)}} \leq \frac{\Delta T_{in,j}}{\tau_{p(j)}} \leq 1 + \frac{\alpha w_p}{\tau_{p(j)}} \quad (3)$$

Where, w_p is pulse width, α is factor selected with model parameter.

The necessary condition of T_{in}, on which the RNN operates, is described as follows:

$$1 - \frac{\alpha w_p}{\tau_0} \leq \frac{T_{in}}{\tau_0} \leq 1 + \frac{\alpha w_p}{\tau_0} \quad (4)$$

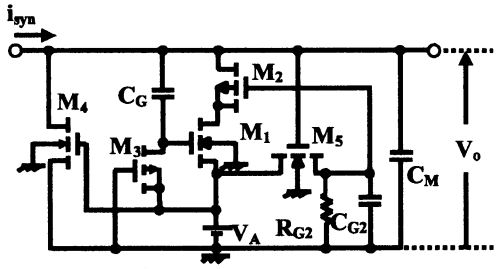


Figure 4: A pulse-type hardware neuron model with an adaptive refractory period.

Figure 4 shows the circuit diagram of the P-HNM with an adaptive refractory period. This figure is composed of two complementary MOSFETs (CMOS) M_1 and M_2, MOS resistors M_3 and M_4, capacitors C_M and C_G, and a voltage power supply V_A. Moreover, as the circuit adjusts a refractory period, we use leak resistor R_{G2}, capacitor C_{G2}, and N-channel MOSFET M_5, as shown in Fig. 4. This P-HNM can adjust a refractory period by changing the time constant $C_{G2}R_{G2}$.

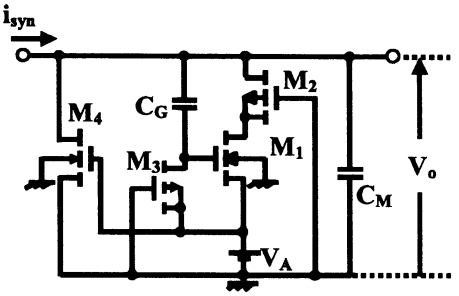


Figure 5: P-HNM.

Figure 5 shows the circuit diagram of the P-HNM used for a neuron model[3]. This figure has CMOS M_1 and M_2, MOS resistors M_3 and M_4, capacitors C_M and C_G, and a voltage power supply V_A.

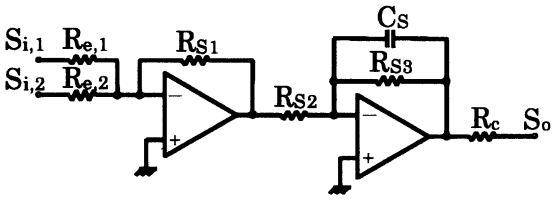


Figure 6: A synaptic part.

Figure 6 shows the circuit diagram of the synaptic part used for a neuron model. As shown in Fig. 6, the adder in the front stage integrates an external input pulse and an output pulse of a neuron model in the front of the synaptic part. Also, the integrator in subsequent stages causes a first-order lag. This circuit has multi-inputs and one output. In this study, the synaptic part is connected to the input of the P-HNM.

In this paper, N_1 of the RNN uses the P-HNM with an adaptive refractory period (Fig.4) and the synaptic part (Fig.6), and $N_{p(j)+1}$ uses the P-HNM with a synaptic part (Fig.5,6), to which a threshold was adjusted so that one pulse might be outputted by two pulses. The remaining P-HNMs with synaptic parts (Fig.5,6) have the ratio of input and output of 1:1 responses.

3 Response characteristics

The circuit parameters of N_1 were as follows: $C_G = 1[pF]$, $C_M = 0.1[pF]$, $M_1, M_2 : W_1/L_1 = W_2/L_2 =$

10, M_3 : $W_3/L_3 = 0.1$, M_4 : $W_4/L_4 = 0.3$, M_5 : $W_5/L_5 = 80$, $V_A = 2.0[V]$, $R_{G2} = 400[k\Omega]$, $C_{G2} = 100[pF]$, $R_{in,1} \ldots R_{in,s} = R_{S1} = 1[M\Omega]$, $R_{S2} = R_{S3} = 10[k\Omega]$, $C_S = 10[pF]$, $R_C = 300[k\Omega]$.The circuit parameters of $N_{p(j)+1}$ were as follows: $C_M = 0.1[pF]$, $C_G = 1[pF]$, $M_1, M_2 : W_1/L_1 = W_2/L_2 = 10$, $M_3 : W_3/L_3 = 0.1$, $M_4 : W_4/L_4 = 0.3$, $V_A = 1.9[V]$, $R_{in,1} \ldots R_{in,s} = R_{S1} = 1[M\Omega]$, $R_{S2} = R_{S3} = 20[k\Omega]$, $C_S = 20[pF]$, $R_C = 300[k\Omega]$. The circuit parameters of remaining P-HNMs with synaptic parts equal the parameters at $N_{p(j)+1}$ except at $V_A = 2.0[V]$.

In this paper, the RNN using the above neuron model were set to $n = 15$, $m = 4$, $p(2) = 2$, $p(3) = 3$, and $p(4) = 4$. In this case, τ_0 was set to $2.3[\mu s]$, and γ was set to $0.9[\mu s]$.

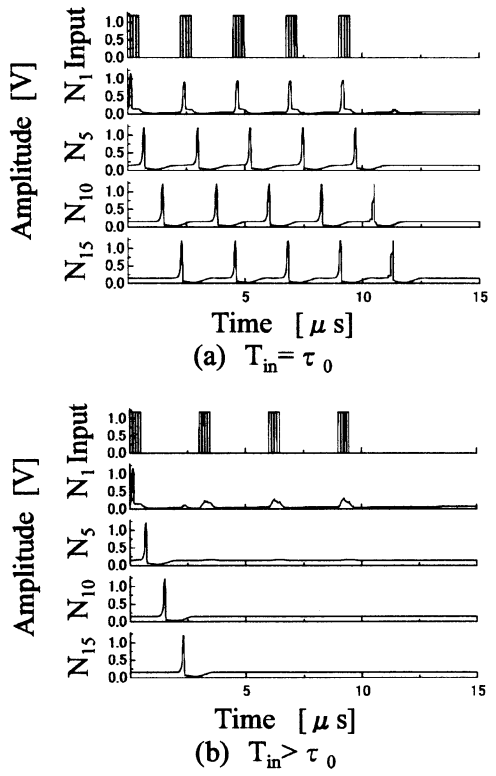


Figure 7: Input-output waveforms.

We discussed the operation of the RNN and its response characteristics.

Firstly, we discussed input-output waveforms to study the operations of the RNN.

Figure 7 shows an example of a simulation result. The input time of a pulse train was $10[\mu s]$ and w_p was $0.1[\mu s]$ and pulse amplitude was $1.2[V]$. The horizontal axis is the time t, and the vertical axes are amplitude of input, N_1 (output), N_5, N_{10}, and N_{15}. Figure 7(a) shows an example of the input-output waveform, where an input pulse train conforms with the pulse train which the RNN memorizes. The number of the pulse train is $\ell = m = 4$, and the cycle period of pulse train is $T_{in} = \tau_0$. This figure shows that the RNN outputs continuous pulse, when T_{in} conforms with τ_0 and N_1 outputs a pulse again. Figure 7(b) shows an example of the input-output waveform, where an input pulse train does not conform with the pulse train which the RNN memorizes. The number of pulse trains are $\ell = m = 4$, and the cycle period of a pulse train is $T_{in} > \tau_0$. This figure shows that a pulse propagates to N_{15}. But, the RNN does not output continuous pulse, when T_{in} does not conform with τ_0, and an external input and output pulse of N_{15} does not input to N_1, which is a relative refractory period concurrently.

As a result, we concluded that the RNN outputs continuous pulses, if the input pulse train conformed to the pulse train, which the RNN memorized.

Next, we discussed the operations of the RNN to the number of pulse trains.

Figure 8 shows an example of the operation characteristic to the number of the pulse trains for different values of ℓ. This figure shows the operation of the RNN, when the construction of the RNN as described above and the cycle period of the RNN is $T_{in} = \tau_0$. The horizontal axis is ℓ, and the vertical axis is the output cycle period T_{out} divided by T_{in}. This figure shows the RNN doesn't output pulse, when ℓ is less than 4, and that the RNN outputs continuous pulse, when ℓ is greater than or equal to 4.

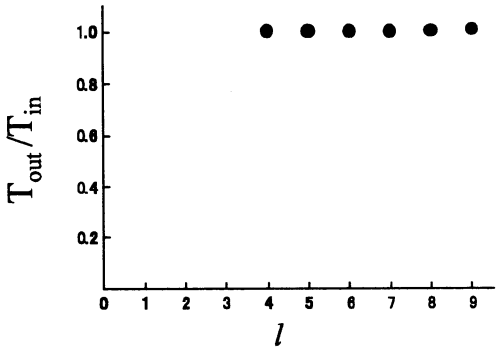


Figure 8: A pulse number characteristic.

As a result, it was seen that the RNN extracted the pulse train and outputs continuous pulses, if the input pulse train comprised of the pulse train, which the RNN memorized.

Furthermore, we discussed the operating range of the RNN for different values of T_{in}.

In this paper, we were set $\alpha = 0.4$, in Eq. (4). Therefore, when the construction of the RNN as shown above, the necessary condition of T_{in}, on which the RNN operates, was described as follows using Eq. (4).

$$0.982 \leq \frac{T_{in}}{\tau_0} \leq 1.018 \quad (5)$$

Furthermore we discussed the operating range of the RNN for different values of T_{in} by circuit simulation using PSpice.

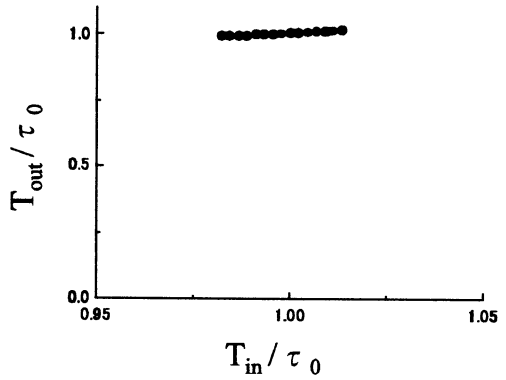


Figure 9: A pulse number characteristic.

Figure 9 shows an example of the simulation result. This figure shows an example of the operating range of the RNN, when all of $\Delta T_{in,j}$ are $\tau_{p(j)}$ and T_{in} is about τ_0. The horizotal axis is T_{in} divided by τ_0, and the vertical axis is T_{out} divided by τ_0. This figure shows the necessary condition of T_{in}, on which the RNN operates, is described as follows:

$$0.982 \leq \frac{T_{in}}{\tau_0} \leq 1.013 \quad (6)$$

In other words, this figure shows that RNN operates when T_{in} is in $-1.8\% \sim 1.3\%$ of range to τ_0. This result approximates Eq. (5).

As a result, the RNN outputs synchronous pulses to a cycle of an external input pattern, if the cycle period of the input pulse train conformed to the natural period of the RNN.

Thus, we verified that the RNN, which was proposed in this paper, can be used as the short-term memory circuit.

4 Conclusion

In this paper, we discussed the construction of a short-term memory circuit using the hardware RNN, and conducted experiments that explain the characteristics of the network through circuit simulation using PSpice.

As a result, we verified that the RNN outputs synchronous pulses to a cycle period of an input pulse train, if an input pulse train comprises the pulse train, which the RNN memorizes, and the cycle period of the input pulse train approximately conforms with the natural period of the RNN. Thus, we verified that the RNN, which was proposed in this paper, can be used as the short-term memory circuit.

In future work, we will discuss the short-term memory circuit, which obtains output when an external input pattern is completely in agreement with the pattern of the memory circuit.

Acknowledgements

This work was supported in part by Grant-in-Aid #14550334 of the Ministry of Education, Science, Sports and Culture of Japan.

References

[1] S.Okada, J.Cao, and S.Tanaka, "A Method for Sustaining Activities in Short-term Memory Model Using Recurrent Neural Network", IEICE Trans. Electron, NC98-89, pp.85-92, 1999.

[2] S.Suenaga, S.Sato, K.Nakajima, "Limit Cycles of a Neural Network with Cyclic Connections and Measurement of an Integrated Neural Circuit", IEICE Trans. Electron, NLP2001-3, pp.13-18, 2001.

[3] Y.Sekine, M.Sumiyama, K.Saeki and K.Aihara, "A Λ-Type Neuron Model Using Enhancement-Mode MOSFETs", IEICE Trans. Electron, Vol.J84-C, No.10, pp.988-994, 2001.

[4] N.Sasano, A.Takeda and Y.Sekine, "A Study on Ring Neural Network Using Pulse-type Hardware Neuron Models with Synaptic Parts", IEICE Trans. Electron, NC2002-78, pp.73-78, 2002.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Statistical Learning based Radial Basis Function Networks for Pattern Classification

JunHyeog Choi*, JungHyun Lee**, KeeWook Rim***

* Division of Computer Science, Kimpo College, KOREA
** School of Computer Science & Engineering, Inha Univ., KOREA
*** Knowledge & Industrial Engineering Dept., Sun Moon Univ., KOREA

Abstract

According to the development of the Internet and the pervasion of Data Base, it is not easy to search for necessary information from the huge amounts of data. In order to do efficient analysis of a large amounts of data, this paper proposes a method for pattern classification based on the effective strategy for dimension reduction for narrowing down the whole data to what users wants to search for. To analyze data effectively, Radial Basis Function Networks based on VC-dimension of Support Vector Machine, a model of statistical learning, is proposed in this paper. The model of Radial Basis Function Networks currently used performed the preprocessing of Perceptron model whereas the model proposed in this paper, performing independent analysis on VD-dimension, classifies each datum putting precise labels on it. The comparison and estimation of various models by using Machine Learning Data shows that the model proposed in this paper proves to be more efficient than various sorts of algorithm previously used.

Keywords: Radial Basis Function, Support Vector Machine, Statistical Learning, Neural Networks.

1. Introduction

There exists many questions before establishing direct pattern classification model from gigantic data[2]. Of those questions, one stems from higher dimension cause. This suggests that when there are too many variables for a target variable which stands for individual label of whole data, there also exists dependency between variables. Through the process, dependency between final pattern classification models triggers multicollinearity which eventually lowers the probability of models[7]. Thus, we can expect better classification results when we are done with proper dimension reduction before performing pattern classification for the data.

RBF Networking has effective capability to deal with both dimension reduction and data classification. This model can carry out unsupervsed learning and supervised learning simultaneously. RBF unsupervised learning is in charge of data dimension reduction.

2. SV-RBF Model

When existing RBF model combines itself with Vapnik's Support Vector Machine, it shows a structure like equation (1). This is called SV–RBF (Support Vector Machine based Radial Basis Function).

$$f(x) = sign\left(\sum_{i=1}^{N} a_i K_\gamma(|x - x_i|) - b\right) \quad (1)$$

In the equation (1), $|x - x_i|$ means the distance between two vectors a_i and b are the parameters to estimate. $K_\gamma(|\cdot|)$ is relying function on distance between two vectors.

Equation (2) shows basic Gaussian function.

$$K_\gamma(|x - x_i|) = \exp\{-\gamma|x - x_i|^2\} \quad (2)$$

To establish equation (2), anyone should decide 4 parameters, x_i the number of N, x_i, a_i, γ.

Even though N, x_i, and a_i are determined by heuristic method, γ is determined by a parameter which minimizes the empirical risk function. So we can choose kernel function $K_\gamma(|x - x_i|)$, which decide inner product. Through this kernel, we build SV-RBF model.

This model can decide 4 parameters automatically, which is different from existing RBF. Table 1 shows this process.

Table 1. Parameter determination of SV-RBF

Parameter	How to decide
N	Number of Support Vector

x_i	Support Vector
a_i	Extended parameter, $a_i = \alpha_i y_i$
γ	Minimize universal function

However, this model shows its limit to multi-pattern classification though it shows excellent capability to binary classification data[13]. In this respect , we need a model that can bring us an appropriate classification result from multi-classification as well as binary classification. This necessity enables us to introduce SL-RBF (Statistical Learning based Radial Basis Function). We decide this model as SL-RBF because it shows superb capability to multi-classification data with stably classified results, compared to SV- RBF.

3. Statistical Learning based Radial Basis Function Networks

Statistical learning is a theory that induces functional dependency from given data. This method contains not only classical statistics such as discriminant analysis, regression analysis and other density estimtion but also machine learning method. Though existing model considers usually the set structure, statistical learning theory defines the relations among variables through learning mchine. This RBF kernel function, shown in this paper, has multi-layer function. RBF below is used in joint function for a hidden layer.

$$H_i = \exp\left(-\frac{(x_1 - C_{1i})^2 + \ldots + (x_k - C_{ki})^2}{\gamma_i^2}\right) \quad (3)$$

Although the number of parameter that has to be estimated in RBF kernel layer perceptron is same as the one in multi-layer perceptron, its way of estimation is rather different. First of all, we estimate the number of center k and radius using unsupervised learning method such as group analysis, then estimate parameter which comes from hidden layer to output layer by using supervised learning method. That is, we use two step estimation–estimating parameter from hidden layer, estimating parameter from outlayer.

In this paper, we apply ZHU's IVM(Import Vector Machine) to RBF first, then use multiple kernel function to determine Import Vector voting for multiple pattern classification. We also determine optimal Import Vector by using multiple kernel function to implement multi-patterns datum classification. IVM can be defined by the equation (4), to which KLR(Kernel Logistic Regression) is applied .

$$-\sum_{i}^{N}[y_i f(x_i) - \ln(1+\exp(f(x_i)))] + \frac{\lambda}{2}\|f\|_{Hk}^2 \quad (4)$$

Equation (4) pursues optimal parameter for a model through Lagrange expansion like SVM[12]. SL-RBF model, shown in this paper, has several steps to follow in order to find an optimal multiple pattern classification.

(STEP 1) Initialization: To determine initial parameter and prameter space

1. $S = \phi, R = \{x_1, x_2, \ldots\ldots x_N\}, k = 1$
   (S : set of import points)
2. Determine the number of RBF function

(STEP 2) Learning & Estimation

1. Determine a parameter to minimize this formula

$$f_l(x) = \sum_{xj \in S \cup \{x_l\}} \alpha_j K(x, x_j)$$

2. Estimate the center of a node group through unsupervised learning method.

(STEP 3) Kernel Bagging & Model Heuristics

1. Find the price x_i^* to meet the condition below

   a. $\arg\min_{xl \in R} H(x_l)$,
   $H(x_l)$ is $y = Sign[f(x)], y_i \in \{-1,1\}$

   b. $average[bagging_{k_{i \in \{k_1, \ldots, k_m\}}} B_{ki}(H(x_l))]$,
   carrying out bagging for multiple kernel function.

2. Determine the width parameter for the center v_j.

   a. Determine nearest distance between centers.

$$\gamma_j = \min_k \|v_k - v_j\|, \text{ for all } k \neq j$$

   b. Determine the parameter to meet the condition 2-a.

$$\alpha_j = \gamma r_j$$

   c. Estimate the weights of a model through 2-a , 2-b.

(STEP 4) Repeat: Repeat (STEP 2) & (STEP 3) to meet the given condition.

SL-RBF Model which this paper suggests is a statistical learning basis RBF perceptron. Multiple kernel function is applied to Zhu's IVM model which depends on logistic kernel function. Kernel functions used in SL-RBF Model are, especially, Running-mean smoothers, Running medians and enhancements, Equivalent kernels, Regression splines, Cubic smoothing splines and Locally-weighted running smoothers.

4. Experimental Results

We compared and calculated the results and misclassification rate according to kernel function pattern in statistical learning model by using Iris data and Arrhythmia data of UCI machine learning repository. It is to have accurate evaluation of efficiency for the suggested algorithm we built in this paper.

First, we calculated the types of hyper-plane and the misclassification rate by classifying data into a random non-linear group under non-linear viewpoint. Then, we compared the results with the ones from the classical ways of analysis such as SVM, SV-RBF, multi-layer perceptron, MLP, and K-Nearest Neighborhood (K-NN) .

Misclassification rate calculation is carried out by using two types of data : training data, test data. We used training data to build a model. We then, used test data to implement classification prediction and to calculate misclassification rate.

We extracted 60% of training data, then saved and used the rest 40% as support and proof data by repeating each experiment over 150 times. Also, we used Matlab ver.6.1, sci-tech programming tool, and a part of R software "The R Project for Statistical Computing".

4.1. Classification experiment through SVM

Iris data is made up of 4 input variables and 1 target variable. 4 input variables determines the outlook of each flower. Besides it has three different labels-Setosa, Versicolor, Viginica- with 150 data. Figure 1 is scatter plot of Iris data over input variables of Petal Width and Petal Length.

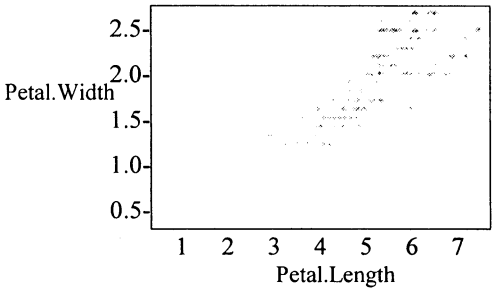
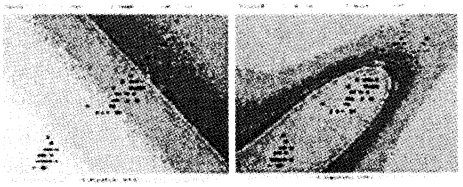


Figure 1. Scatter plot of Iris data

First, we experimented classification analysis for the two groups - Versicolor, Verginica - to implement SVM-adopted classification experiment. Then, we did classification analysis for Setosa.



a) Linear b) Gaussian

Figure 2. Classification plane of SVM by Kernel function

Figure 2 is a visual expression for the classification model when linear kernel function and Gaussian function are used. As seen in Fig. 2, we can find differences in effectiveness according to the types of kernel function. It shows us that Gaussian kernel function is a better classification model than linear kernel function.

Table 2 suggests a summary for Fig. 2.

Table 2. Performance according to Kernel function(Iris)

	Linear	Gaussian
Margin	0.1356	0.0492
Number of Support vector	11	13

According to fig 2, Gaussian kernel function shows less margin and bigger number of support vector which shows better explanation of a model than linear kernel function. As a next step, we performed experiments by using Arrhythmia data.

This data has 279 input variables and 1 target variable which has 16 labels. We performed same experiments as we did for the Iris data. Table 3 shows the results of the experiments. It says that 452 data are used in the experiments. Table 3 suggests our classification rate is much higher when Gaussian kernel function is used than when earlier linear kernel function is used .

Table 3. Performance according to Kernel function(Arrhythmia)

	Linear	Gaussian
Margin	0.1692	0.0538
Number of Support Vector	19	24

4.2. Compared experiments between proposed model & earlier classification models

We evaluate classification efficiency by comparing earlier models like K-NN and MLP with SL-RBF and SV-RBF model. Table 4 shows the results.

Table 4. Misclassification ratio of five models

Model	Iris	Arrhythmia	Average
Gaussian	0.2132	0.2316	0.2224
SV-RBF	0.1232	0.1501	0.1367
K-NN	0.2942	0.3262	0.3102
MLP	0.5621	0.6138	0.5880
SL-RBF	0.0843	0.1123	0.0983

Table 4 shows misclassification rate of several classification method for Iris data and Arrhythmia data. It also shows the mean of misclassification rate for the two types of data - Iris & Arrhythmia. The results show that SL-RBF has the least misclassification rate.

5. Conclusions and Future Work

In this paper, we suggested SL-RBF Model – statistical learning radial basis function - which shows superb classification results for multiple classification data as well as binary classification data. The model seen in this paper brings lower misclassification rate through repetitive experiments.

Performing supervised learning and unsupervised learning all together, however, can bring about unexpected cost problem. Cost efficiency and time efficiency are the things we should consider in the future. Jackknife method can be very useful to deal with efficiency problems.

6. Acknowledgement

This research was supported in part by a grant from the 2003 Kimpo College.

7. References

[1] C. Burges, "A tutorial on support vector machines for pattern recognition," In Data Mining and Knowledge Discovery, 1998.

[2] V. Cherkassky, F. Mulier, "Learning from Data: Concept, Theory, and Methods," John Wiley & Sons, Inc., 1998.

[3] T. Evgeniou, M. Pontil, T. Poggio, "Regularization networks and support vector machines," MIT Press, 1999.

[4] P. Green, B. Yandell, "Semi-parametric generalized linear models", Proceedings 2nd International GLIM Conference 1985.

[5] G. Kimeldorf, G. Wahba, "Some results on Tchebycheffian spline functions", Math. Anal. Applic. 1971.

[6] X. Lin, G. Wahba, D. Xiang, F. Gao, R. Klein, B. Klein, "Smoothing spline ANOVA models for large data sets with Bernoulli observations and the randomized GACV," Technical Report998, Department of Statistics, University of Wisconsin, Madison 1998.

[7] R. H. Myers, "Classical and Modern Regression with Applications," Duxbury, 1990.

[8] M. J. D. Powell, "The theory of radial basis functions approximation in 1990," Advances in Numerical Analysis Volume II: Wavelets, Subdivision Algorithms and Radial Basis Functions, W. A. Light, ed., Oxford University, pp. 105-210, 1992.

[9] A. Smola, B. Scholkopf, "Sparse Greedy Matrix Approximation for Machine Learning," In Proceedings of the Seventeenth International Conference on Machine Learning 2000.

[10] V. N. Vapnik, "The Nature of Statistical Learning Theory," New York: Springer Verlag, 1995.

[11] V. N. Vapnik, "Statistical Learning Theory," New York: Wiley, 1998.

[12] G. Wahba, "Support Vector Machine, Reproducing Kernel Hilbert Spaces and the Randomized," GACV. Technical Report 984rr, Department of Statistics, University of Wisconsin, Madison 1998.

[13] J. Zhu, T. Hastie, "Kernel Logistic Regression and the Import Vector Machine," NIPS2001 Conference 2001

[14] http://www.ics.uci.edu/~mlearn/MLSummary.html

[15] http://www.r-project.org/

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Evolution of Hybrid Neural Networks using Genetic Network Programming

Dazi Li
Department of Electrical and Electronic Systems Engineering
Kyushu University, JAPAN

Kotaro Hirasawa Jinglu Hu
Graduate School of Information Production and Systems
Waseda University, JAPAN

Kiyoshi Wada
Department of Electrical and Electronic Systems Engineering
Kyushu University, JAPAN

Abstract

Genetic Network Programming (GNP) is a methodology for evolving arbitrary directed graph programs consisting of a special network with nodes, that has been successfully applied to a variety of problems. In this paper, a new approach to the construction of hybrid neural networks by means of GNP is presented. A random search method(RasID) is combined to adjust the network's connection weights. This paper describes the approach, operators and results of the application of the model to several problems and show the flexibility of GNP to evolve solutions with different architectures for a given problem.

1 Introduction

The evolution of neural networks has been widely studied since 1980s. Feed-forward neural networks can be trained and constitute a good solution for many difficult problems once their topology is known. Unfortunately, the right topology is problem dependent and usually not easy to find. The evolutionary systems based on Genetic Algorithms or Genetic Programming mainly focus on the evolution of network's topologies and weights[1]. However, little work has been performed concerning different types of neurons. Our wok concentrates on the evolution of the hybrid neural networks with different neuron types using GNP as an alternative.

Conventional Feed-forward neural networks consist of pure sigmoidal neurons. However, scientists have found that multiplication also plays a very important role in the neural computing system[2]. A multiplication unit is defined, which can be combined with sigmoidal units to form different structures of neural networks depending on the problems. Our former study shows that this multiplication unit can act as a very important role in the function approximation problems when it is applied to the network topology with uniform convergence on compact sets.

There are many methods to find a better structure of a network for a problem. Constructive algorithms, pruning method, so called optimal brain damage and a method using higher order derivatives[3] were also used to selectively remove connections while maintaining network accuracy. In this paper, an evolutionary technique called Genetic Network Programming (GNP) has proven useful in finding an optimal hybrid network structure.

The remainder of this article is organized as follows: section 2 will give a brief overview of (GNP). A simple description for hybrid neural networks and RasID learning algorithm will be given in Section 3. Evolution of hybrid neural networks is discussed in section 4. Section 5 will include the experimental results and section 6 will provide conclusions.

2 Genetic Network Programming

Genetic Network Programming(GNP) is a kind of evolutionary methods, which evolves arbitrary directed graph programs. A program in GNP is composed of a network with nodes, which gives the name to our approach. In this section, we briefly describe the structure and behavior of GNP. For a more detailed description see[4].

2.1 GNP General Structure

A GNP is constituted of:

- A set of nodes: start node, judgement node and processing node.
- A set of connections, which determines the structure of a program.

A GNP is run by sequentially executing the programs attached to nodes in it. Firstly, the start node is activated; Then the flow of the control are transferred to the next node, that may be judgement node or processing node. A judgement node does predefined judgement and then transfers the control to the

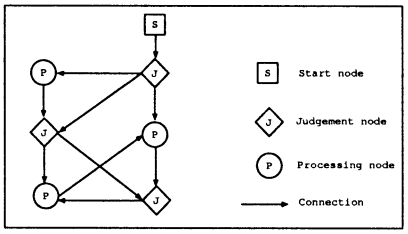


Figure 1: A simple depiction of GNP structure

next nodes indicated by the connection. A processing node does some predefined processing, then transfers the control to the next node indicated by the connection(see Figure 1).

2.2 GNP Operators

A GNP is also a population based evolutionary strategy, and the chromosomes in GA correspond to the programs associated with GNP's nodes. The three operators used in GA are also used to evolve GNP.

GNP Mutation - Mutation changes the nodes and their connections, so that a randomly chosen node and connection of each program are substituted by a new randomly selected connection.

GNP Crossover - The crossover operator combines two selected individuals and generates two offsprings. A uniform crossover is applied to GNP. In GP, crossover is executed by exchanging sub-trees in the individual, however, GNP crossover can exchange "subprograms" between different individuals by swapping the connected nodes and their connections.

GNP Selection - Mutation or crossover happens in an environment where the individual is selected. Some evolutionary methods use a simple function of the fitness measure to select individuals, which is called fitness-proportionate selection. Tournament selection uses a model in which certain randomly selected individuals compete in a subgroup and the fittest is selected. The evolution also takes elite strategy to keep the best individuals of one generation to the next generation.

The desired behavior for a GNP is to solve some problems posed by the environment. To do this, the information provided by the environment can be seen as its inputs and outputs can be used as instructions to eventually achieve a solution. This is achieved by executing the programs associated with the network's nodes. Programs associated give the node's output based on the input available to it, as a result, control the flows of information between nodes.

3 Hybrid Neural Networks

The concept of hybrid neural networks was proposed in [5]. Hybrid neural networks can be constructed by different neural units, i.e, multiplication units and summation units are combined in the networks. There can be many different combinations of summation units and multiplication units to construct different solutions for problems.

3.1 Main Components

Conventional summation unit acts as the basic component of Feed-forward neural networks. The net input into a unit is generally calculated by the weighted sum of the outputs from the previous unit. The output of unit j in a network can be described as:

$$h_j = f_j(\alpha_j);$$
$$\alpha_j = \sum_{i \in JF(j)} w_{ij} h_i + \theta_j. \quad (1)$$

where w_{ij} is the weight parameter from node i to node j; $JF(j)$ is the set of suffixes of nodes which are connected to node j; θ_j is the threshold parameter of node j. Function $f_j(\cdot)$ that governs the operation of the nodes can be any continuously differentiable functions, typically sigmoidal function or hyperbolic tangent function can be employed.

Differently from the summation unit described above, in spite of combining all the inputs by a linear summation, multiplication unit developed multiplies all the inputs after subtracting an adjustable weight from them. Output of multiplication node j can be described as:

$$h_j = f_j(\alpha_j);$$
$$\alpha_j = z_j \prod_{i \in JF(j)} (h_i - w_{ij}) + \theta_j. \quad (2)$$

where z_j is the gain parameter of node j. Activation function $f_j(\cdot)$ can be linear function. Multiplication units are found more effective and biologically similar to real neural systems than conventional summation units.

3.2 Random Search with Intensification and Diversification

The gradient-based schemes such as well-known back-propagation(BP) algorithm are well used for neural network training. However, while the multiplication units provide an advantage of an increased information processing capacity compared to summation units, a major drawback of multiplication units is an increased number of local minima.

In this paper, we train the hybrid neural network which is got by evolution of GNP using a modified random search method called RasID(Random search with intensification and diversification). RasID executes intensified and diversified search in a unified manner using information on success and failure of the past searching. For details, see reference[6].

4 Evolution of Neural Topologies

We developed a GNP based evolutionary method of neural networks in an attempt to find an optimal architecture of a hybrid neural network for a given problem.

4.1 Network Representation

For application of GNP to topology optimization, a new element, "memory" is introduced. The memory consists of two parts: inputs($IN_1, IN_2, \cdots, IN_m$) and networks($Net_1, Net_2, \cdots, Net_n$). Length of the input part m is decided by the number of inputs, that of the network part n should be decided beforehand. The relation between GNP and memory was depicted in Figure 2.

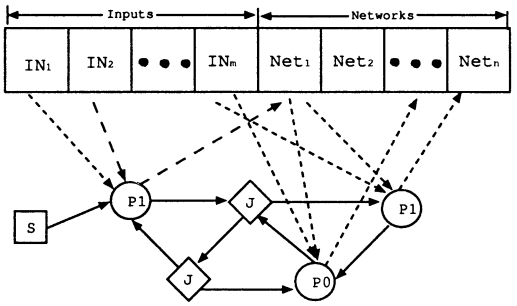


Figure 2: An example for GNP genotype to neural network mapping

Before executing the GNP, all the network part of memory is randomly initialized to a simplest network with only one input node. After executing the start node of GNP, judgement nodes and processing nodes will be activated sequentially. Judgement node executes a certain judgement, such as the usage frequency of this node, and decides which node to activate next. There are two kinds of processing nodes: summation node with sigmoidal activation function and multiplication nodes with linear activation function. Connections exist between the GNP's processing nodes and memory: two connections from the memory to the processing node, and one connection from the processing node to the memory. Processing node takes two networks from the input part or network part of the memory, and processes the two networks as subnets, that is, connects the two networks using one neuron, while the type of the neuron is designated by the type of processing node. The network pointed by connection from the processing node to the memory will be renewed by the generated hybrid network.

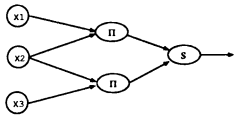


Figure 3: A hybrid neural network representation of the GNP in Figure 2.

After running GNP, many networks are constructed in the memory. Figure 3 shows a representation of a hybrid network by GNP. The network has 3 inputs, two hidden neurons and one output neuron. The neurons in the network are multiplication unit(π) and summation unit(S) respectively.

4.2 Evolving GNP

An initial population is composed of N individuals. All individuals have exactly the same number of GNP nodes. During the dynamic creation of neurons in the network, a restriction is made on the memory's maximum depth. While this depth is not reached, GNP will continue to execute the programs. Another restriction is the time to execute one GNP. The execution of one GNP will stop whenever the maximum depth is reached or the pre-determined time is over. The time is calculated by summing the time consumed executing different nodes, for example, the execution of judgement node will take time t_j while that of processing node is t_p. Generally we defined like $t_p > t_j$. The GNP individuals can create neural networks of different sizes and shapes because many hybrid networks are stored in the memory of GNP. The evolution process can be described as follows:

Do while n < generations {
  Create an initial population of GNP
  Execute all GNP individuals and generate homologous hybrid neural networks
  Training each neural network using RasID on the training set
  Select individuals depending on the fitness
  Generate new GNP population in the next generation using crossover and mutation
}

Dynamically reconfiguring hybrid networks is the key characteristic of the evolution of hybrid networks using GNP. Because of the compact genotype representations of the GNP, that is, fully connected and repeated nodes, there is no need for large GNP structure to represent the hybrid network for the problems.

5 Results

In order to evaluate hybrid networks evolved by GNP, computational experiments were carried out.

All the experiments that follow use a population size of 10 GNP individuals for the evolution. In the experiments, we used GNP with tournament selection(tournament size=2) and also crossover and mutation probabilities of 80% and 10%, respectively. The weights and biases were randomly initialized within the range $[-1, 1]$. Each hybrid network was trained for 100 iterations using RasID. This value may differ according to different problems. For complex problems, this number may be larger so that network training has sufficient time to converge. If a lower number is used, and training is stopped too early, hybrid network's full potential to learn a data set may not be accurately tested. For each problem, we performed 10 runs with different random seeds.

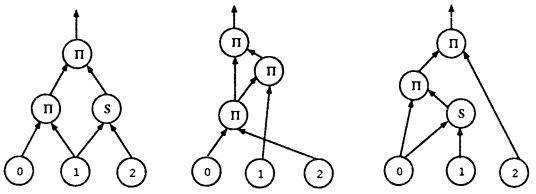


Figure 4: Decoded solutions obtained for XOR.

To compute the fitness, we calculated the average sum of the squares of the errors between the desired outputs and actual activation outputs of the hybrid network. This is the measure for the supervised learning whose most implementations are trying to minimize. Table 1 shows the results obtained for XOR and parity 3 problems: the generations to converge, the number of hidden neurons and probability to get solution correctly. Many different structures of hybrid neural networks have been constructed for these problems. Three different results for XOR problem are shown in Figures 4.

Table 1: Summary of results for XOR and parity 3 problems.

task	generations min/avg/max	hidden neurons min/avg/max	solutions
XOR	3/9.7/24	2/3.3/6	100%
Parity-3	6/17.8/36	4/6.5/11	80%

6 Conclusions

GNP's network structure makes the nodes be used repeatedly and polymorphic structures are obtained. The results show that the algorithm can search any architecture for hybrid networks which is suitable for the given problem, so that the algorithms converge very quickly for simple problems such as XOR and parity 3. These properties improve the flexibility of GNP evolved to the fittest solution for a given problem.

Acknowledgments

This research was partly supported by the 21st Century COE Program. "Reconstruction of Social Infrastructure Related to Information Science and Electrical Engineering."

References

[1] X. Yao, "Evolving Artificial Neural Networks", *Proceeding of the IEEE*, vol. 87, no. 9, pp 1423-1447, 1999.

[2] C. Koch and I. Segev, "The role of Single Neurons in Information Processing," *Nature Neuroscience Supplement*,Vol. 3, pp. 1171-1177, 2000.

[3] K. Hirasawa, S. Kim, J. Hu, J. Murata, M. Han, C. Jin, "Improvement of Generalization Ability for Identifying Dynamical Systems by Using Universal Learning Networks", *Neural Networks*, Vol.14, pp.1357-1404, 2001.

[4] H. Katagiri, K. Hirasawa, J. Hu and J. Murata, "Network structure oriented evolutionary model-genetic network programming and its comparison with genetic programming", *2001 Genetic and Evolutionary Computation Conference, Late Breaking Papers*, pp. 219-226, 2001.

[5] D. Li, K. Hirasawa, J. Hu and J. Murata, "Hybrid Universal Learning Networks", *IEEJ Trans. EIS*, Vol.123, No.3, pp. 552-559, 2003.

[6] J. Hu, K. Hirasawa, and J. Murata, "RasID-Random Search for Neural Network Training", *J. of Advanced Computational Intelligence*, vol.2, no.4, pp.134-141, 1998.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Auditory Evaluation Experiments of Japanese Vowel Synthesized by Using Cascaded Sand-glass Type Neural Network

M. Kimoto T. Shimizu H.Yoshimura K. Sugata M. Tanaka
Tottori University, Faculty of Engineering, Department of Information and Knowledge Engineering
4-101 Koyama-Minami, Tottori city, 680-0945 Japan

Abstract

We proposed a new scheme to derive the characteristics of vowels from LSP parameters as well as to compress information by using cascaded sand-glass type neural network(nonlinear, 5-layers)(CSNN(NL5)). Here, it was shown that LSP parameters can be restored with well feasibility for practical use of speech synthesis from the compressed LSP parameters. In order to verify their audibility, we performed auditory evaluation of synthesized speech by using artificial parameters on the hidden layer output plane. As a result, we verified the compressed LSP parameters by our scheme have workable quality for speech synthesis.

1 Introduction

In the study of speech analysis, formants are useful characteristics to extract the information of speech signals. The formants are superior components of speech in the frequency domain. Vowels have three distinctive formants in many cases, and they are named "first formant" F_1, "second" F_2, and so on. It was known that five Japanese vowels /a/, /i/, /u/, /e/, /o/ are ranged pentagonally on F_1-F_2 plane.

The formants are adequate parameters for speech analysis to characterize vowels. However, it is not easy to calculate formants through computer algorithms. Linear predictive parameters in frequency domain such as LSP(Line Spectrum Pair) parameters[1] were employed so far to encode and decode speech signals. LSP parameters have advantageous properties that give us not-complicated algorithms to calculate them by computer.

We devised a new scheme to derive the characteristics of vowels from LSP parameters as well as to compress the information by using cascaded sand-glass type neural network(CSNN)[2], [3]. Here, it was shown that CSNN is a very capable network to analyze and synthesize speech signals. The analyzed information of LSP parameters by CSNN classifies vowels which is similar to the result on F_1-F_2 plane. CSNN can also synthesize speech by decompressing. Sand-glass type neural network(SNN), named from it's shape resemblance, was proposed as a convenient network to compress the information of pictures by Cottrel et al.[4]. SNN is a very preferable neural network since we do not need to prepare, in advance, teacher signals for learning process. Input signals themselves should be teacher signals to compress the information.

In this paper, we discussed the quality of speech synthesized from compressed LSP parameters by our scheme. In order to verify their audibility, we performed auditory evaluation experiments of synthesized speech from the compressed LSP parameters. In consequence, we showed that the results of auditory evaluation gave good agreement with clustering distribution of the compressed parameters of vowels. CSNN worked out adequately to decompress LSP parameters and synthesized speech.

2 Cascaded Snad-glass type Neural Network

2.1 Outline of the Structure of Cascaded Sand-glass type Neural Network CSNN

Fig.1 shows the structure of CSNN(NL5) for compressing LSP parameters. As shown in Fig.1, CSNN(NL5) has the structure of cascading SNN(NL5)s, called network unit SNN(NL5). Here, when we have to mark precise indication of the attribute of SNN, we give notation such as (NL5), which indicates SNN consists of 5 layers and both second and 4th layers are made of non-linear neural units. The first layer is for input of LSP parameters, second layer is made of non-linear neural units for the information processing to compress, the third is hidden layer of a linear neural units, 4th layer is made of nonlinear neural units to decompress, and 5th layer is for output. In this way, the first, second, and hidden layers work out cooperatively for the information compression whereas hidden, 4th, and 5th for the information decompression.

The hidden layer has a single neural unit. CSNN consists of several cascaded SNNs which are first stage SNN, second stage SNN and so on. The total learning process of CSNN is performed at several consecutive stages SNNs. Hence, each stage has an output signal of its hidden layer. These output signals constitute a compressed signal of LSP parameters.

The input signal to the first stage SNN of CSNN is LSP parameters. Thus, first stage SNN learns LSP parameters. The second stage SNN learns error signal of

first stage's error which is obtained by subtracting the outputs of first stage from the inputs. The third stage SNN learns error signal of second stage's error which is obtained in the same way. Each stage learns error signal of preceding stage SNN. The output of CSNN, as a whole, is the total amount of outputs of each stage SNN. When learning procedure of CSNN is completed, the hidden neural unit of each SNN can output the principal component of input signals.

2.2 CSNN(NL5) employed in this Investigation

There are two types of neural unit. One is linear unit whose response function is linear and the other is non-linear unit whose response function is non-linear. Here, we adopted sigmoid function. Both input layer and output layer consist of 14 linear neural units in accordance with the analysis order 14 of input signal through LSP parameters.

The single neural unit of hidden layer is linear. All the neural units of the second and 4th layer are non-linear. What is essential to analyze input speech signal and compress the information is how many neural units should be arrayed in the second and 4th layers. This number was found to be crucial to enhance the ability of encoding and decoding by CSNN(NL5). This number had been determined by various experiments to be 20, which endowed with preferable abilities. We adopted Recurrent Least Squares(RLS) algorithm[5] for the learning process of network unit SNNs. Hereafter, it is called RLS-learning, which brings about fast convergence of SNN's learned errors. The two outputs from two hidden layers of first stage unit SNN and second stage SNN constitute the axes of distribution plane. It was found by experiments that the distribution resembled to the distribution of F_1-F_2 formants analysis.

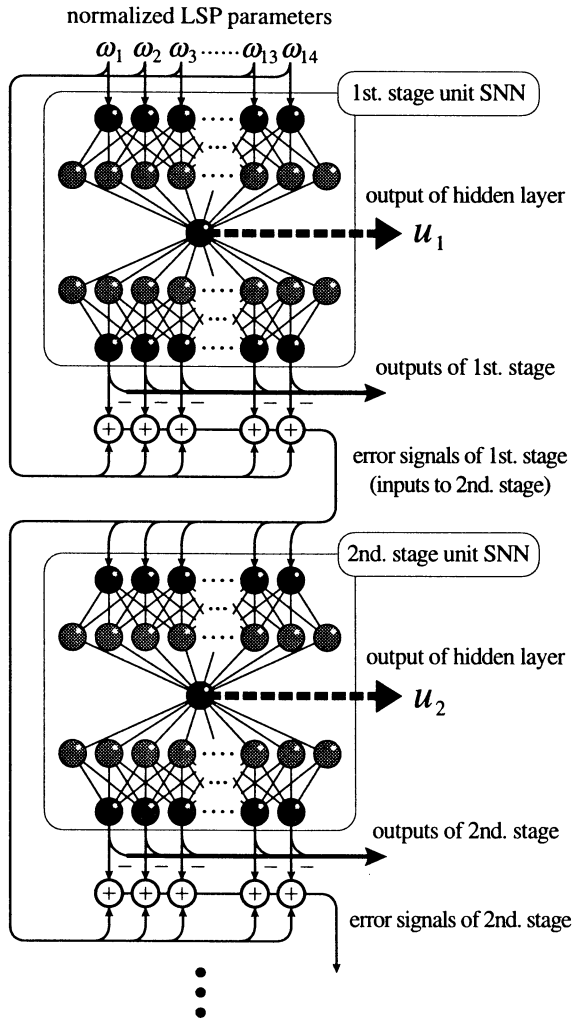


Fig.1 Structure of CSNN(NL5).

3 Learning CSNN(NL5)

3.1 Data Set for Learning CSNN(NL5)

We recorded five Japanese vowels uttered by four male native Japanese speakers for learning data set to CSNN(NL5). The utterance was digitized by 16-bits resolution with a sampling rate of 11.025kHz. We applied LSP analysis shown in Table 1. The length of LSP parameters is 200 frames cutting out the stable part from each utterance. The data set for learning CSNN includes 4,000 LSP parameters(200 frames × utterance ×5 vowels×4 speakers). These were arranged randomly

Table 1 Specification of unit SNN.

A/D conversion	
sampling frequency	11.025 kHz
quantization bits	16 bits/sample
LSP analysis	
analysis order	14-order
frame length	256 samples(23.22 msec)
frame interval	128 samples(11.61 msec)

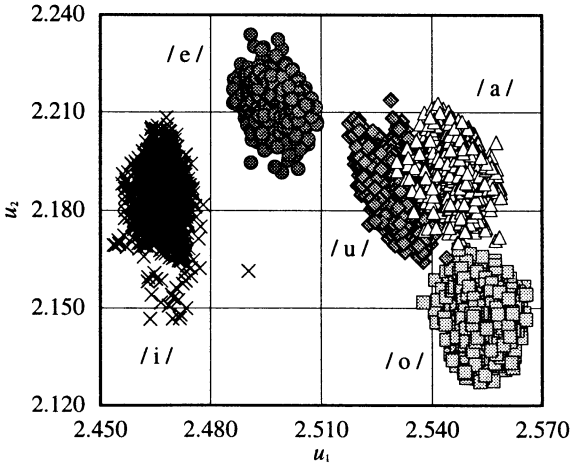


Fig.2 Example of outputs from hidden units.

to be input sequence. LSP parameters are normalized in the value range from 0.1 to 0.9 to input to the CSNN.

3.2 Learning CSNN(NL5)

The outputs error of CSNN(NL5) decreased rapidly when the RLS-learning method was used. It was found by the preliminary experiments that the output error of learning process converged to be small enough after three times input sequence(4,000 parameters × 3 times = 12,000 iterations). In the following experiments, to allow for a margin, we repeated learning process of SNN by 120,000 iterations(4,000 parameters × 30 times). The initial weights of SNN were chosen randomly to have values ranged from -0.01 to 0.01.

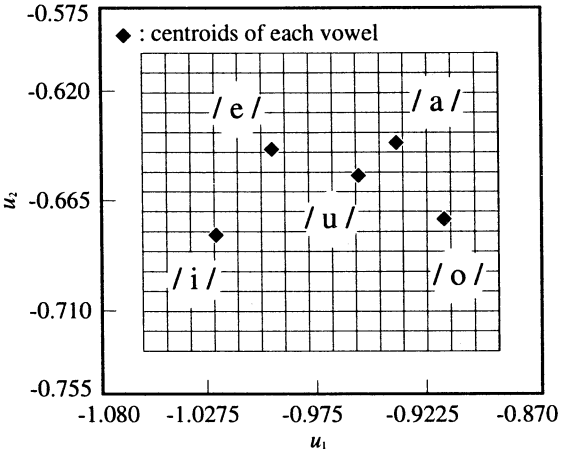


Fig.3 Hidden layers outputs from 2 CSNN(NL5).

4 Auditory Evaluation Experiments

4.1 Speech Synthesis Using Parameters Chosen Artificially from Output Plane

We evaluated the square error of the decompressed LSP parameters from the inputted LSP parameters. We showed LSP parameters can be restored with well feasibility for practical use of speech synthesis from the compressed LSP parameters. In this paper, for the purpose of verifying audibility of synthesized speech which was generated from the parameters pointed out from the hidden layer output plane(Fig.3), we performed auditory evaluation experiments. The parameters were pointed out in order from 256(16×16) lattice points of the hidden layer output plane(u_1-u_2 plane). The 256 lattice points cover up the whole domain of compressed LSP parameters of Japanese vowels. We chose 256 lattice point values of u_1-u_2 plane as input parameters to hidden neural units of CSNN(NL5). CSNN(NL5) decompressed to restore 256 LSP parameters which were used for the following auditory evaluation experiments.

4.2 Auditory Evaluation for Synthesized Speech

We adopted 256 LSP parameters restored above and used excitation pulse of 120Hz pitch frequency to synthesize speech. The 19 experimental subjects were males and females around twenty years. We ask them to listen to the synthesized speech and to choose one answer

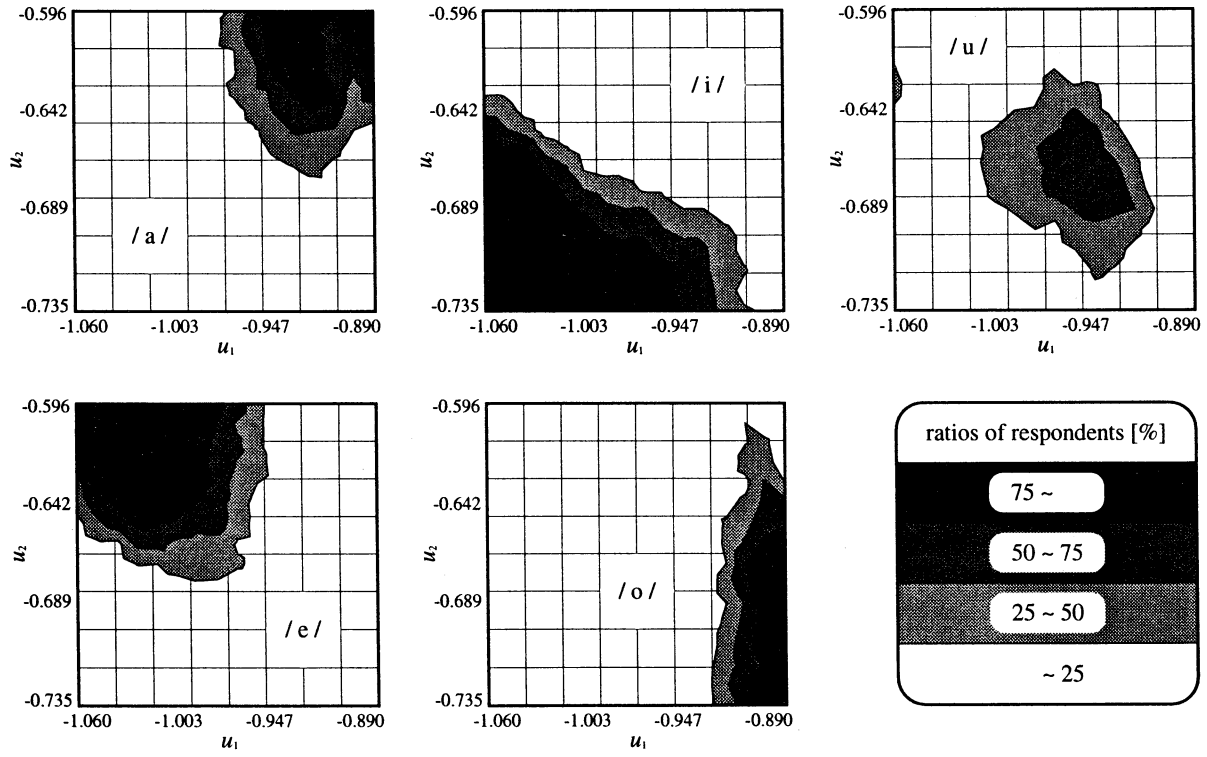


Fig.4 Results of auditory evaluation.

from 6 choice of /a/, /i/, /u/, /e/, /o/ and otherwise. Fig.4 gave contour curves of the respondents ratios of /a/, /i/, /u/, /e/, /o/. Here, the ratio of respondents of otherwise was less than 5% anywhere at lattice point of u_1-u_2 plane. Thus, its contour curve was abbreviated in Fig.4. Thus, we can say each vowel has its own auditory domain separately on u_1-u_2 plane.

4.3 Relation between Distribution of Hidden Layers Output and Auditory Domain on u_1-u_2 Plane

Fig.3 and Fig.4 were summarized to be Fig.5 in order to compare the distribution of compressed LSP parameters of each vowels with auditory domains on u_1-u_2 plane. Fig. 5 took up into consideration 50% or more domain from Fig.4 and the centroids of each vowels on Fig.3. Fig.5 made us assured that auditory domains plotted artificially have accordance with the distribution of compressed LSP parameters. Speech signals synthesized from parameters around the centroids of each vowels domain on u_1-u_2 plane were audible with more than 75% score. Thus, CSNN(NL5) was shown to be a capable neural network for speech analysis and synthesis with use of LSP parameters.

5 Conclusion

We performed auditory evaluation experiments of synthesized speech with use of CSNN(NL5). We verified each vowel has its auditory domain on hidden layer's signal plane. The synthesized speech was audible well enough to be practical application.

References

[1] N. Sugamura and F. Itakura, "Speech Data Compression by LSP Speech Analysis-Synthesis Technique," Trans. of IEICE, vol.J 64-A, no.8, pp.599-606, 1981 (in Japanese)

[2] T. Shimizu, H. Yoshimura, T. Namiki, N. Isu, and K. Sugata, "Analysis of Japanese vowel by using sandglass type neural network," IEICE Technical Report, SP2000-162, pp.17-24, 2000 (in Japanese)

[3] M. Kimoto, T. Shimizu, H. Yoshimura, N. Isu, and K. Sugata," Auditory Evaluation of Japanese Vowel Synthesizied by Using Cancaded Sand-glass Type Neural Network," Technical Report of IEICE, SP2002-173, pp.23-28, Jan. 2003 (in Japanese)

[4] Cottrel G. W., Munro P., and Zipser D., "Image compression by back-propagation : An example of extensional programming," Advances in Congnitive Science, 3, pp.208-240, Ed. Sharkey N.E., Norwood, NJ : Ablex, 1988

[5] Scalero T. S. and Tepedelenlioglu N., "A fast new algorithm for training feedforward neural networks," IEEE Trans. Signal Process., 40, 1, pp.202-210, 1992

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A neural network-based remote guidance interface for rescue robot

Zhixiao Yang*, Kazuyuki Ito**, Kazuhiko Saijo*, Akio Gofuku**, Fumitoshi Matsuno***
* Graduate School of Natural Science and Technology, Okayama University
3-1-1 Tsushima-naka, Okayama 700-8530 Japan
{y.zhixiao/k.saijo}@usm.sys.okayama-u.ac.jp
** Department of Systems Engineering, Faculty of Engineering, Okayama University
3-1-1 Tsushima-naka, Okayama 700-8530 Japan
kazuyukiito@ieee.org fukuchan@sys.okayama-u.ac.jp
*** Department of Mechanical Engineering and Intelligent Systems, University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan
matsuno@hi.mce.uec.ac.jp

Abstract It is difficult to navigate a rescue robot with full automation under unknown environment. We developed a semi-automation and remote manipulation robot system to execute rescue tasks by novice users. In the system, the robot can move automatically, or be guided by a remote operator. The remote part plays an important role in doing rescue tasks. We developed a force feedback control-monitor interface for novice users to guide the robot remotely. Although the interface provides support of vision, hearing and touch for human, it is not good for catching the focus of a user's attention to the screen. In order to enhance the efficiency of doing tasks and reduce the time spent on finding survivals, we developed a neural network-based mouse-screen remote guidance interface as the supplement of the previous one. With the interface, a user can guide the robot just by clicking the screen image. He need not frequently move his attention out of the environment image. In this approach, a back propagation neural network is constructed to translate the 2D image into 3D world.

Key words interface, rescue robot, remote guidance, BP neural network

1. Introduction

Disaster sites are often unreachable and too dangerous for human exploration. Human rescuers have very short time to find trapped victims in a collapsed structure; otherwise the chance of finding victims still alive is nearly zero. So it is necessary to develop robot systems, which in place of human execute rescue tasks. Disaster sites where human cannot reach have unknown construction. So the robot should have the ability to doing rescue tasks by it self. For example, it can know what is a target, how to access to the target, how to avoid an obstacle. In other words, the robot must have some extend of automation, or intelligence. In fact, even to let a robot do a task that is very simple for human, such as to go to a goal or avoid an obstacle, is very difficult. It is also unable to let a robot execute a task with full automation in an unknown construction. To overcome this problem, we developed a semi-automation and remote operation robot system, which is presented in [1]. Through the remote operation, a manipulator can share the control to do a task with the robot. Thus when the robot cannot do the task correctly, human can help it.

The system was designed as figure 1. A camera is mounted on the tank like robot. The environment information, such as that of roads, walls, obstacles and goals, is transmitted to a computer and displayed on the monitors. The connection between the camera and monitors is wireless. An operator can guide the robot remotely while seeing the environment information on the screen.

In our system, the remote part plays an important role during the course of the whole process. In such unstructured and continuously changing environment, the rescue robot has to be guided to collect information mainly by a remote operator. In conventional research, the main efforts are put into the development of robots. Although to develop robots is important, it has no reason

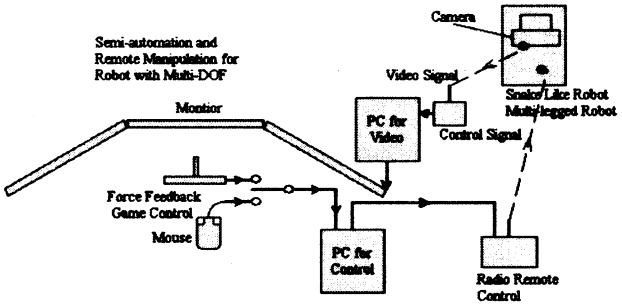


Fig.1 The outline of the rescue robot system

to ignore the essentiality of the interface design between the users and the robot. Because for the remote user, the interface is the only way he can interact with the robot. The interface design for the rescue system should be fully considered.

For rescue tasks, time is one of the most important factors. The shorter the time spent before finding a victim, the higher the probability he or she would survive. So the interface should be propitious to improve efficiency. The system is design for any user, whether he is a robotics expert, or a novice. The interface should be easy to learn to operate without special knowledge. It should also be able to fill human's needs and support their limitations, such as the vision of eyes, the touch of hands, and the hearing of ears, etc. In a word, an interface should be humane [2]. Thus, the designed interface would make the user not only do tasks, but also enjoy it.

2. The force feedback control-monitor interface

In [1], we use a force feedback control as the

interface between the user and the robot, since the action of the tank like robot is similar to that of a car. Like figure 2, the control has one wheel and two pedals, which are in charge of direction and speed respectively. When a user operates the control, the control computer gets its output through a USB connector and transfers it into a value of voltage in binary. Then the voltage value is passed to a radio emitter through a D/A board. The robot responses when it receives the radio signal.

Fig.2 The force feedback control guidance

The force feedback control provides a good operation interface for a user, since many users have the experience with driving a car. Even if the user has never driven cars, it is easy for him to learn and to operate. However, when one is operating the control, he often has to focus his attention on driving the robot. For example, where to turn left, how much extent to turn, where to speed and where to brake. In other words, he has to make the path plan and track it by himself. Sometimes in accessing a special location, he may experience waste instructions and has to back and drive again. Thus the efficiency of searching victims would reduce and the time before finding them would postpone. So, as a supplement, we developed another remote guidance method.

3. The mouse-screen interface

While doing remote operation, it is the monitor that catches a user's attention mostly. The operator has to spend most of the time on watching the screen, to distinguish if there is a victim or not, if there is an obstacle, which path is good for navigation. It would be better for a user to do remote operations without having to move his attention out of the screen. Using the screen and the mouse as the interface, we developed the second remote guidance method. When a user makes out a goal on the screen, he can drive the robot just by clicking it with the mouse. The left work, such as path planning and tracking, will be executed by the system automatically. People who use computers know that when he wants to click a position on the screen, he can use the mouse easily, nearly without attention transferring. Such an interface is good to obtain high efficiency.

The key to this approach is to know the goal's actual position in the real world, basing on its screen image. It can be obtained through the optical knowledge or using range sensors, such as the GPS device, the sonar or laser range finders [3 4]. The former method is a little complex. It needs many parameters such as the height of the camera to the ground, the angle between the camera's axis and the ground, the focus position of the camera, etc. Even to measure the angle of the camera's axis to the ground is difficult. The latter method would expand the cost and bring more tasks such as to filter the noise, to analyze the echoed data, etc.

The artificial neural network has ability to infinitely approximating any mapping, linear or nonlinear. It can also generalize the obtained mapping from the training set to non-training space. A well-constructed neural network would be appropriately to get the mapping from an image to the real world.

3.1 The mouse-screen interface outline

The outline of the mouse-screen interface is shown in figure 3. When a user recognizes a goal through the monitor, he clicks it. The goal's image position on the screen is inputted into an artificial neural network. The network processes the data and outputs the real position

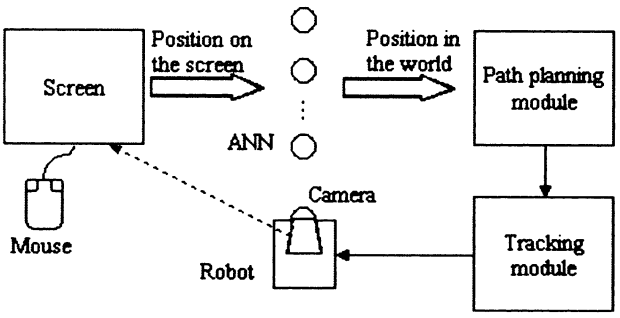


Fig.3 The outline of mouse guidance

of the goal in the world. Then the data is transmitted to the path-planning module. This module generates a proper path to access the goal. Then the tracking module drives the robot to the object location along the path.

3.2 The Cartesian coordinate systems for the screen and the world

The first step is to create Cartesian coordinates for the real world and the image respectively. It is easy to set the coordinates for an image. For a point in the world, three coordinates are needed. If we use three coordinates to localize an object, the difficulty of training set sampling for neural network will increase, and the structure of the neural network will be very complex. In order to lighten workloads and simplify the neural network's structure, we give an assumption.

3.2.1 Assumption

It is assumed that the robot and the objects those can be detected before it locate at the same plane, that is, the ground. This is true in many cases. The valid guidance range for one image is about 80cm. Generally, the terrain does not change suddenly in a small area. It can be

thought that objects on the ground of a small area are in the same plane. For the case that they are not in the same plane, such as the case that the goal locates at the top of a slope but the robot locates at the ground, we will discuss in 3.4. With the assumption, the dimension of the world's coordinates reduces to two.

3.2.2 The coordinates for the screen

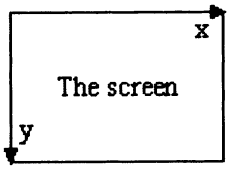


Fig.4 the coordinates for screen

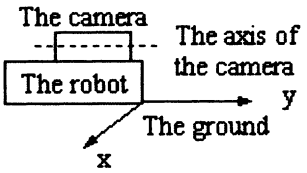


Fig.5 the coordinates for the world

The coordinates for the screen are indicated in figure 4. The horizon and vertical of the screen are set as axis x and axis y respectively. The active direction of axis x is right. The active direction of axis y is downward. The origin is the left up of the screen. The coordinates are measured in pixels.

3.2.3 The coordinates for the world

The coordinates for the world are indicated in figure 5. Axis y is the projection of the camera's axis on the ground. The active is the forward direction of the robot. The origin is the intersection point between axis y and the projection of the robot's front side on the ground. The active direction of x is the robot's right side. The coordinates are measured in centimeter.

3.3 The neural network-based 3D position translation

In our system, we used back propagation neural network [5 6]. The algorithm adjusts neuron connection weight using the criterion of the global output error. When the global error is bigger than the expected error, the algorithm calculates and adjusts the weight again. When it is smaller than or equals to the expected error, it is said that the network converges. Back propagation means that the calculation begins from the output layer, then one by one, to hidden layers.

At first, the training set is carefully selected. A board is made, on which we draw some girds. Like figure 6, parallel lines are drawn every 10cm along the transverse and lengthwise direction respectively. The intersection points of these lines can be chosen for training the neural network. The points those are drawn as black solid circle are chosen as the output vector of the training set. With a wide view camera, a 800×600(in pixels) image can be got, like figure 7. The position of every point corresponding to the real world can be obtained. Thus the input vector of the training set forms. In our training set, there are 107 input-output pairs. With the created training set, we can construct the structure of the neural network with the simulation tool of Matlab [6]. At last, the neural network is constructed with one input layer, one output layer and two hidden layers. The first hidden layer has 4 neurons, while the second one has 6. The hidden layers use sigmoid activating function, while the output layer uses linear one. The constructed network converges quickly and the global error goes under 0.1.

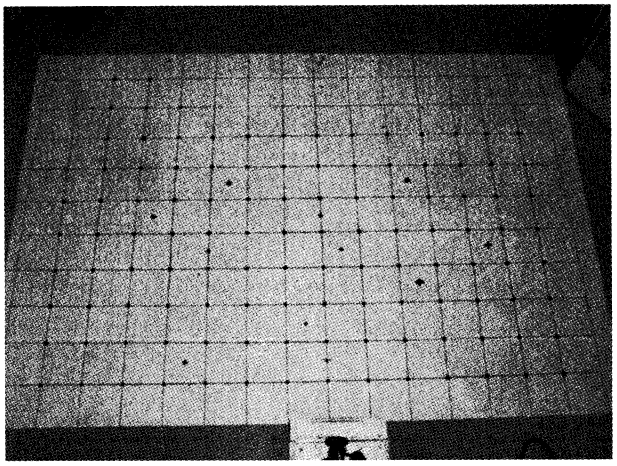


Fig.6 The grid board

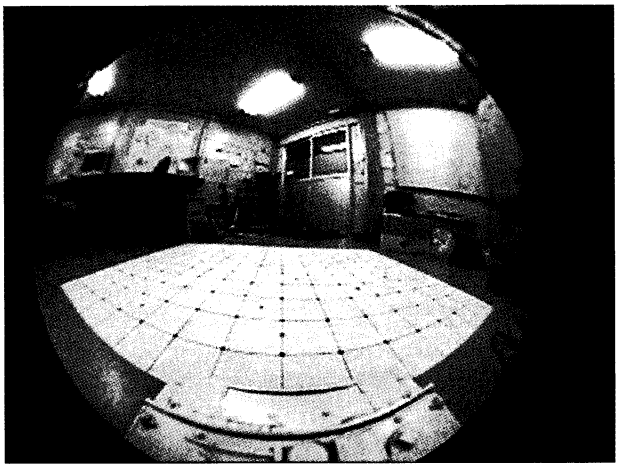


Fig.7 The image of the grid board

11 input-output pairs that are different from the training set are used to test the trained neural network. In figure 8, the points marked with asterisks denote the inputs of the training set. The points marked with squares denote the inputs of the testing set. In figure 9, the outputs corresponding to the inputs of the training set and the testing set are indicated with points that are marked with crosses and triangles respectively. The output of the neural network shows that, compared with the expected output, the maximal errors of x and y are 1.1(cm) and 1.9(cm) respectively, whose relative errors are 4.4% and 2.9% respectively. Errors may occur when the training and testing data cannot be read precisely. The precision of the network will be improved when we overcome these problems. Although this, the result is satisfied with our purpose. Figure 10 shows the outputs of the neural network corresponding to the input vector of the testing set and the expected outputs, which are real positions in the world. The points marked with triangles denote the neural network's actual outputs. The points marked with circles denote the expected outputs. The figure shows that the neural network's outputs nearly

cover the points indicated by the expected positions.

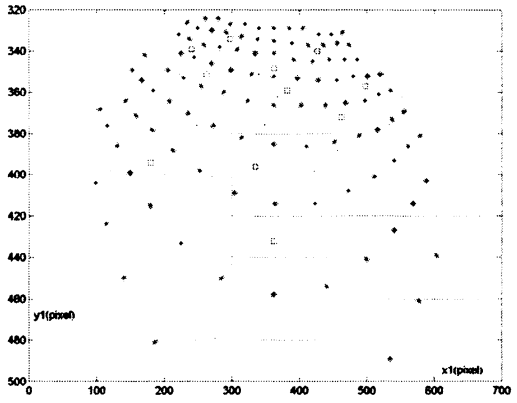


Fig.8 The inputs of the network

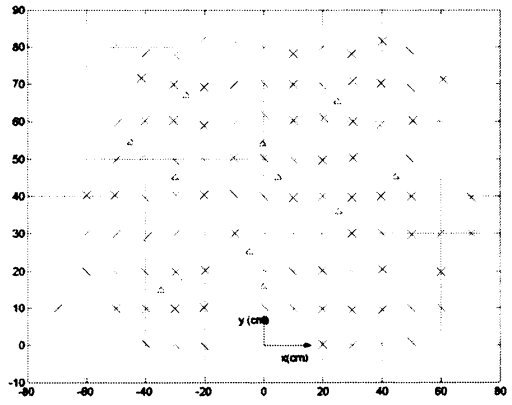


Fig.9 The output of the network

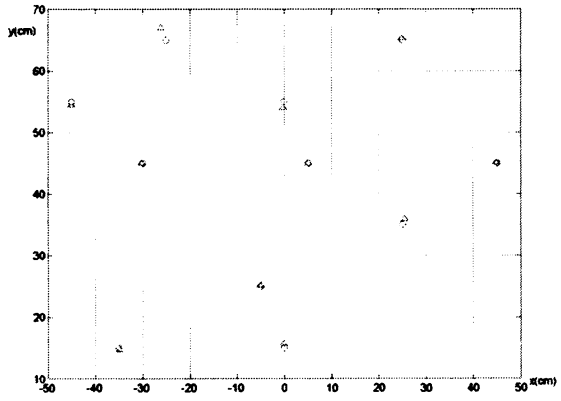


Fig.10 The expected and actual network

3.4 Discussion

The camera's pose and height compared to the ground are not permitted to change after training the neural network. Thus the obtained mapping between the screen image and the world will continuously be valid during the course of doing tasks.

In 3.2, it is assumed that the robot and goals are in the same plane. For the case that the terrain varies suddenly, the neural network-based guidance method is inapplicable. The user can shift to use the force feedback control guidance.

4. Conclusion

In this paper, as a supplement of the force feedback control-monitor interface that was described in [1], we developed an artificial neural network-based mouse-screen interface to guide the rescue robot remotely. In this interface, an operator can guide the rescue robot by clicking the goal position's image on the screen. While doing tasks, the user does not have to move his focus of attention from the monitor frequently. He can put his main effort into finding goals in the vision displayed on the screen. Thus his efficiency will enhance and the time spent on finding survivals would reduce. To realize the translation of locations from 2D screen image to 3D real world, a back propagation neural network is constructed. The test result shows that, the network can give satisfying output. Using this interface, one can even guide the robot to a very distant range by dividing it into many short ones step by step.

Acknowledgements

This research was performed as a part of Special Project for Earthquake Disaster Mitigation in Urban Areas in cooperation with International Rescue System Institute (IRS) and National Research Institute for Earth Science and Disaster Prevention (NIED).

References

[1] Kazuyuki Ito, Zhixiao Yang, Kazuhiko Saijo, et al. "Development of a remote controlled robot with intelligent human-interface for collecting information (in Japanese)," the papers of Technical Meeting on Nuclear Energy, IEE Japan, NE-03-8, p17-21, 2003

[2] Jef Raskin, The humane interface: new directions for designing interactive systems, ACM Press, 2000

[3] K.Nagatani, H.Ishida, S.Yamanaka, et al. "Three-dimensional localization and mapping for mobile robot in disaster environment," Proceedings of IROS 2003, pp.3112-3117

[4] Yasuyoshi Yokokohji, Masarnitsu Kurisu, Saida Takao, et al. "Constructing a 3-D map of rubble by teleoperated mobile robots with a motion canceling camera system," Proceedings of IROS 2003, pp.3112-3117

[5] Martin T. Hagan, Howard B. Demuth & Mark Beale, Neural network design, the University of Colorado Press, 1996

[6] Shuang Cong, Neural network theory and application with MATLAB toolboxes (in Chinese), University of Science and Technology of China Press, 2003

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Growing neural network with hidden neurons

Ryusuke KURINO, Masanori SUGISAKA and Katsunari SHIBATA
Dept. of Electric and Electronic Engineering, Oita University
700 Dannoharu Oita 870-1192 Japan. Email: shibata@cc.oita-u.ac.jp

Abstract

The authors have proposed "growing neural network" in which axons of neurons grow according to the concentration gradient of a chemical substance that are diffused from the output neurons in proportion to the error signal. In this paper, we propose a new technique that enables to obtain an appropriate structure including hidden neurons. The hidden neurons diffuse another chemical substance when they don't have enough connections with the input neurons. In a simulation, it was examined that an appropriate structure could be formed through the learning of the most popular and simplest non-linear separation problem of "EXOR".

1. Introduction

Artificial neural networks (NN) are broadly used to approximate non-linear functions because of its advantage of the learning and generalization ability. When we use the artificial NN for function approximation, three-layer structure is usually employed. The reason is that it is difficult to decide an appropriate structure adaptively for a given problem. Moreover, it is well known that 3-layer neural network with enough number of hidden neurons can approximate any continuous functions with any precision[1][2].

However, in the 3-layer NN, each hidden neuron represents just a linear and smooth classification of the input space, and it is difficult to represent complicated abstract information. On the other hand, in our humans, high order functions, which include recognition, memory, and control, are supported by the complicated structure of NN in the brain. Such functions are realized as a various levels of abstract state representation, the authors think, and it is very useful for the effective learning based on generalization. Useful abstract state representation can be formed in the hidden neurons when a series of processing from sensors to motors is acquired through learning such as reinforcement learning. That must be the origin of intelligence by which the humans are distinguished from modern robots, the authors believe. To realize an intelligent robot with such functions, autonomous acquisition of an appropriate structure of the NN including recurrent structure will be required.

It is known that in the brain of living things, the number of neurons decreases after its birth, while the number of connections between neurons increases rapidly[3]. This suggests a strategy in the brain as follows. More neurons than necessity are prepared at first, then they grow their axons and learn their synapse weights, and an appropriate network structure is formed flexibly. After that, unnecessary neurons are removed by apoptosis. It has been also reported that chemical substances such as NGF(Nerve Growth Factor) make a role of promoting the growth of axons[3].

The authors have proposed the growing NN that introduced the concept of growth of neuron into the conventional NN as an extension of learning[4]. Getting a hint from NGF mentioned above, axons grow according to the concentration gradient of a chemical substance which is diffused according to the error signal propagated in BP (Back Propagation) learning. Since the error signal is diffused as the chemical substance, it is expected that an appropriate structure for a given problem can be obtained. As the first step, simple logical functions "AND" and "OR" were learned by the network, and two-layer structure could be obtained.

However, by two-layer structure, it is well-known that only liner separation problems can be learned. In order to solve the other type of problems that need non-linear operation, hidden neurons are necessary. The hidden neurons cannot make connections to the other neurons by the following two reasons. (1) The hidden neurons cannot grow its axons appropriately since the hidden neurons do not have the information of the input signals. (2) The input neurons cannot grow its axons to the hidden neurons since the hidden neurons do not have the information of the error signal. In order to form the network with hidden neurons, it is necessary for the hidden neurons to make connections with the input or the output neurons by some method.

In this paper, we propose a new technique that enables to obtain an appropriate structure including the hidden neurons. The hidden neurons diffuse another chemical substance not depending on the error signal and ask for the connections from nearby input neurons when they don't have enough connections with the input neurons. In a simulation, it is examined whether an appropriate structure can be formed through the learning of the most popular and simplest non-linear separation problem of "EXOR".

2. Growing NN with hidden neurons

2.1 Fundamental algorithm

In the growing NN with hidden neurons, the growth is formulated as an extension of BP learning. Fig.1 shows the idea of the growing NN with the hidden neurons. Fig.2 shows the flowchart of the algorithm. At first, the output and the error are calculated, and the error signal is propagated backward. If

the output neuron does not have enough connections, the neuron diffuses the signal as a chemical substance. Then, the concentration gradient is formed by the diffusion around the output neuron. The growing neuron extends its axon according to the concentration gradient.

The hidden neurons that do not have enough connections diffuse another substance constantly not depending on the error. By the diffusion of the hidden neurons, the input neurons extend its axons and make connections to the hidden neurons at first. After that, the weight is increased gradually until the hidden neuron makes a connection to an output neuron. The hidden neuron that has formed the connections with the input neurons extends its axon to an output neuron referring to the input signals. From just after making the connection (synapse), the learning according to the regular BP algorithm begins.

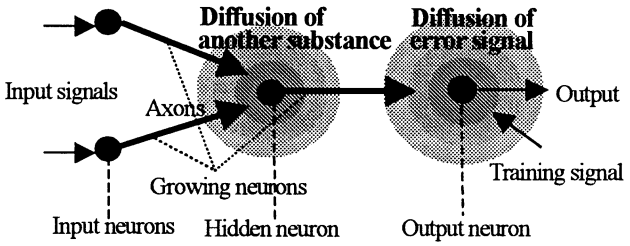


Figure 1: Growing NN with hidden neurons.

1. Decision of each neuron's position.
2. Decision of whether each connection has been made or not.
3. Setting of input and training signals.
4. Calculation of output and error.
5. Back propagation of the error signal.
6. For the neuron that can get the error signal:
   Diffusion of the error signal.
   For the other neurons: Diffusion of the other substance.
7. Calculation of the axon growth.
8. Judge of whether each connection has been made or not.
9. Update of connection weights(Only for connected neurons.)
10. Return to step 3.

Figure 2: Flowchart of the growing NN.

2.2 Diffusion

In the growing NN, the error signal is diffused as a chemical substance around the output neuron. Same as the previous paper[4], the substances for the positive error and that for the negative error are prepared. The error signal is calculated as

$$\delta_j = -\frac{\partial E}{\partial net_j} = (d_j - o_j)\cdot f'(net_j) \qquad (1)$$

where j = 0,..,$NODE_j$, $NODE_j$: the number of the output and the hidden neurons, δ_j: the error signal, net_j: the internal state of the neuron j, o_j: the output, d_j: the training signal, $f'(net_j)$: the derivative of the output function, and E: the error function. There are three kinds of substances here, and the concentration of each substance is denoted by u^p, u^n or u^h where p denotes "positive error", n denotes "negative error" and h denotes "from hidden". The diffusion of each substance is calculated according to the diffusion equation as

$$\frac{\partial u_{x,y}}{\partial t} = div + D\nabla^2 u_{x,y} \qquad (2)$$

where $u_{x,y}$: the concentration at (x,y), div: the divergence of the substance and D: a diffusion constant. At the place of the output neuron, when $\delta > 0$, $div = \rho\delta$ (ρ: a divergence constant) for u^p, and when $\delta < 0$, $div = -\rho\delta$ for u^n. Otherwise, $div = 0$. At the place of the hidden neurons, $div = \alpha$ (α: a divergence constant) for u^h, and $div = 0$ for u^p and u^n. At all the other places, $div = 0$ for all the substances.

2.3 Extension of the axon

The axon of a growing neuron extends according to the state of the neuron and the concentration gradient at the tip of the axon. Two types of neurons are prepared here. One of them called "positive neuron" makes only a positive connection. While, the other called "negative neuron" makes only a negative connection. Here, in order to make the growth of the axon smooth, the first-order delay A_i is introduced to the growth of neurons. The extension of the axon is calculated as

$$\tau_a \frac{dA_i}{dt} = -A_i + (\nabla(u^p - u^n)\cdot flag_i + \nabla u^h)\cdot S_i \qquad (3)$$

$$\frac{da_i}{dt} = \xi \cdot A_i \qquad (4)$$

where τ_a: a time constant, a_i: the position vector of the tip of the axon, ξ: a growth constant and i = 0,...,$NODE_i$, $NODE_i$: the number of the growing neurons that include the input and hidden neurons. The time constant is large so as to calculate the temporal average of the growth of the axon. For the positive neurons, $flag$ = 1, while for the negative neurons, $flag$ = -1. The state of a growing neuron S is defined as the first-order delay of the output of the neuron. The first-order delay is introduced to adjust the delay of the error signal with a time constant τ_s. Moreover, since the hidden neurons could not extend its axon by the concentration gradient of a self-diffusion ∇u^h, the effects of the diffusion are removed in Eq (3) only for the hidden neurons.

2.4 Update of the connection weight

A connection weight is always 0 before the connection is formed. When the connection is formed, the synapse begins the learning from 0 connection weight. The update of the weight is calculated as well as the regular BP leaning as

$$\frac{dw_{ji}}{dt} = -\eta \cdot \frac{\partial E}{\partial w_{ji}} = \eta \cdot \delta_j \cdot o_i \qquad (5)$$

where w_{ji}: the connection weight from the neuron i to the neuron j and η: a learning constant.

If the hidden neuron has not made the connection to the output neuron yet when the input neurons formed the connection to the hidden neuron, the neuron cannot update the weight according to the error. Then, the weight is increased monotonically and gradually not depending on the error. The weight change is calculated using an exponential function as

$$\frac{dw_{ji}}{dt} = \gamma \exp^{-\lambda \sum_j |w_{ji}|} \cdot flag_i \qquad (6)$$

where γ: a learning constant and λ: a constant that influences the slope of the exponential curve. The reason why the exponential function is used is to avoid that the weight becomes too large. Moreover, when the weight becomes less than 0 in a positive neuron, the weight is set to 0. When it becomes more than 0 in a negative neuron, it is set to 0. When the hidden neuron does not have the connection to the output neuron, biases are changed with a large time constant τ_θ so as that the temporal average of the neuron's output becomes the middle value 0.5. Otherwise, they are calculated by BP learning.

3. Simulation

3.1 Set up

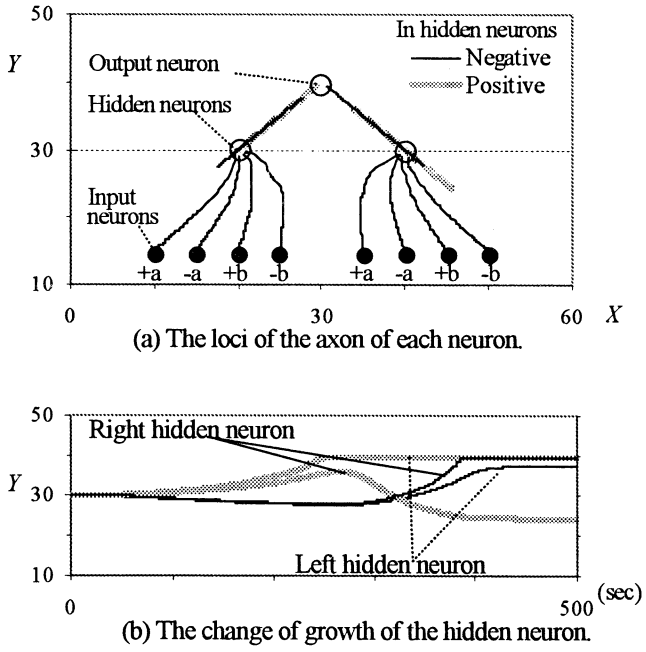
By using the growing NN with the hidden neurons, the non-linear separation problem "EXOR" was learned. The simulation is done on 0.6(mm)× 0.6(mm) of plane. The concentration is calculated by difference on each point on a 61× 61 grid. There are eight input neurons. The position of each neuron can be found in Fig.3. All the neurons do not have any connections beforehand. Among the input neurons, the input '+a' and '+b' indicate the positive neurons, the input '-a' and '-b' indicate the negative ones. It is assumed that each positive hidden neuron and one of the negative hidden neurons are located at the same position, and the input neuron makes the connections with the both neurons simultaneously. The input and the training pattern can be seen in Table1. The patterns are presented in this order and each one is presented for 1(sec). The calculation as shown in Fig.2 is done for every 0.001(sec). The output function is sigmoid ranges from 0 to 1. The parameters used in the simulation are as follows. The diffusion constant D = 0.1(mm^2/sec), the divergence constant ρ = 10.0(1/sec), α = 0.15(1/sec) for hidden neurons, the growth constant ξ = 0.1(mm^2/sec), the time constant τ_a = 10 (sec) τ_s = 0.1(sec), τ_θ = 10 (sec), the leaning constant η = 0.2(1/sec), γ = 0.001(1/sec), the constant λ = 0.15.

3.2 Results

Fig.3 shows how the axon of each input or hidden neuron grows. At first, the axon of each input neuron extends according to the diffusion of the hidden neurons. Since the diffusion is constant, the axon of each input neuron continues to grow towards the closer hidden neuron. The inputs '-a' and '+b' that is close to the hidden neurons formed the connections earlier. The loci of the axons of the input neurons are curved by the influence of the other hidden neuron. After all the input neurons made the connections to the hidden neurons, the positive hidden neurons grew its axons, and the negative hidden neurons grew in the opposite direction at first. The left positive hidden neuron formed the connection to the output neuron at 246(sec). After this connection, the right positive neuron inverted the direction of the growth, and the negative hidden neurons grew to the output neuron. Then the right negative hidden neuron formed the connection to the output neuron at 387(sec). After this connection, the left negative hidden neuron and the right positive one stopped to grow.

Fig.4 shows the change of the error. The arrows in the figure indicate the timing when a hidden neuron formed the connection to the output neuron. Fig.5, 6, 7 show the change of the (input - left positive hidden), (input - right negative hidden) and (hidden - output) connection weights respectively. Table1 shows the error for each input pattern roughly. Error (a) indicates the error when there is no connection between the hidden and output neurons. Error (b) indicates the error after the left positive hidden neuron made the connection to the output.

When there is no connection between the hidden and output neurons, the error decreases only by the learning of the bias of the output neuron. When the input pattern changes from (1) to (2), the output is still 0.1 since the bias does not change instantaneously. Then, the positive error appears as shown in Table1. However, the error decreases soon by the learning of the bias. When the input pattern changes from (3) to (4), the negative error appears since the output is still 0.9.



(a) The loci of the axon of each neuron.

(b) The change of growth of the hidden neuron.

Figure 3: Growth of each neuron.

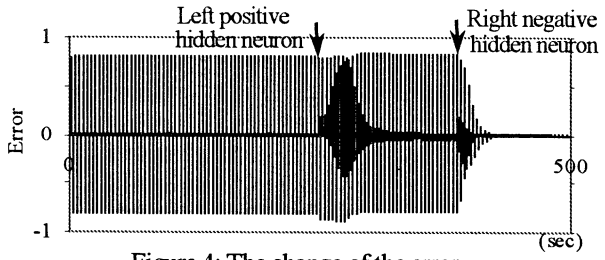


Figure 4: The change of the error.

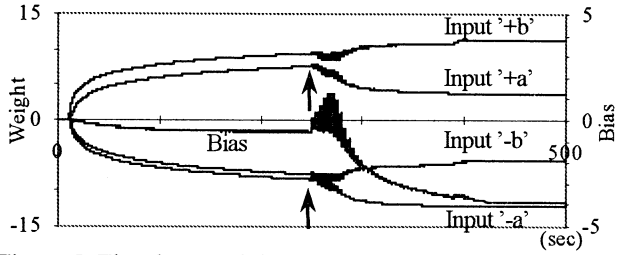


Figure 5: The change of the weight between the input and the left positive hidden neurons.

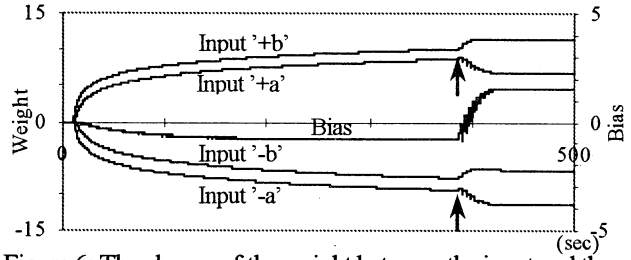


Figure 6: The change of the weight between the input and the right negative hidden neurons.

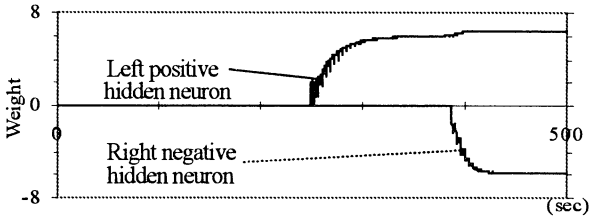


Figure 7: The change of the weight between the output and the hidden neurons.

Table1: The training signal and the error for each input pattern.

Pattern	Input'a'	Input'b'	Training	Error(a)	Error(b)
(1)	0	0	0.1	0	0
(2)	0	1	0.9	Positive	0
(3)	1	0	0.9	0	Positive
(4)	1	1	0.1	Negative	Negative

When each hidden neuron formed the connection to the output neuron, the learning of the connection weight according to the error signal begins. As shown in Fig.5, because the left positive hidden neuron made the connections with the input '-a' and '+b' earlier, their weights are larger than those of the input '+a' and '-b'. Accordingly, in case of the error (b), when the input pattern changes from (1) to (2), the input '+b' changes from 0 to 1. Then, the output of the hidden neuron increases and the output becomes larger than 0.1. Thus, for the pattern (2), the positive error decreases and finally becomes almost 0. On behalf of it, when the input pattern changes from (2) to (3), since the input '-a' changes from 0 to 1 and the input '+b' changes from 1 to 0, the output of the hidden neuron decreases and the output becomes less than 0.9. Thus, the positive error increases, and that promotes the right negative hidden neuron to grow its axon toward the output neuron. Moreover, the negative error increases slightly for the pattern (4) as shown in Fig.4 even though the connection from left positive neuron has been formed. When the input '+b' changes from 0 to 1 on Table1, the output of the hidden neuron increases and becomes larger than 0.9, and the negative error increases slightly. After the right negative hidden neuron formed the connection, the error finally becomes almost 0. Accordingly, the connection weight converged, and the left negative hidden neuron and the right positive one stopped to grow.

After the learning, the left positive hidden neuron was excited only for the pattern (2) since its bias of the hidden neuron is negative. The right negative hidden neuron was excited for the pattern (1), (2) and (4) since the bias of the hidden neuron is positive. The output neuron was inhibited by the right negative hidden neuron for the pattern (1) and (4), but was excited by the positive bias of the output neuron. The bias that is not shown in the figures is 2.4.

4. Conclusion

In this paper, a new technique that enables to obtain an appropriate structure including the hidden neurons was proposed. When the hidden neurons do not have enough connections, the hidden neurons diffuse the substance not depending on the error signal constantly. In a simulation, the input neurons formed a connection to the hidden neurons at first, and then three-layer structure could be obtained through the learning. There still remain many problems to be solved, for example, parameter tuning is necessary to modify the connection weight in the hidden neurons that do not have enough connections to the output neuron.

Acknowledgement

This research was partially supported by the Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research, #14350227 and # 15300064.

Bibliography

[1]K.Hornik, Multiayer Feedforward Networks are Universal Approximators, Neural Networks, Vol.2, pp. 359-366(1989)

[2]K.Funahasi, On the Approximate Realizaton of Continuous Mappings by Neural Networks, Neural Networks, Vol.2, pp.183-192 (1989).

[3]T.Tsumoto, Brain and Development, Asakura Publishing Co.,Ltd (1986). (in Japanese)

[4]R.Kurino, M.Sugisaka, K.Shibata, Growing neural network for acquisition of 2-layer structure, Proc. of IJCNN'03, pp.2512-2518 (2003).

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Backpropagation Algorithm for Rice Yield Prediction

[1]Puteh Saad, [1]Mohd Rizon M Johari, [3]Nor Khairah Jamaludin, [2]Siti Sakira Kamarudin, [3]Aryati Bakri, and [1]Nursalasawati Rusli

puteh@kukum.edu.my, rizon@kukum.edu.my, miss_khairah@hotmail.com, sakira@uum.edu.my, aryati@fsksm.utm.my, nursalasawati@kukum.edu.my

[1]Northern Malaysia University College of Engineering (KUKUM), Taman JKKK, Kubang Gajah, 02600 Arau, Perlis, Malaysia. Telephone : 604-9798285 Fax : 604-9798292.
[2] School of Information Technology, Universiti Utara Malaysia, Sintok, Kedah, Malaysia
[3]Faculty of Computer Science and Information System, University Technology Malaysia, Skudai, Johor, Malaysia

Abstract

Parameters that affect rice yield are many, for instance diseases, pests and weeds. Statistical or mathematical model is unable to describe the correlation between plant diseases, pests and weeds on the amount of rice yield. In this study, a Backpropogation (BP) algorithm is utilized to develop a neural network model to predict rice yield based on the aforementioned factors in MUDA irrigation area, Malaysia. The result of this study shows that the BP algorithm is able to predict the rice yield to a deviation of less than 0.21.

Keywords: BP, rice yield prediction, improved unit range.

1.0 Introduction

Accurate early estimates of the final yield are essential for the determination of management information and to evaluate the results of management strategies in agriculture. Thus it is important to consider the accuracy of the prediction methods.
Statistical methods are generally used to explain the spatial variations and predict crop yield at field scale[4]. However statistical methods require a normal distribution of input variables, which is not always the case [5]. Many complex mathematical models have been developed for modeling crop yield [2]. Although these models produce excellent results, they require numerous input variables that are time consuming and expensive to obtain in the field. BP algorithm is proven to alleviate the limitations mentioned by the following observations. It has successfully predicted corn yield based on soil texture, topography, pH and some soil nutrient element [3]. The prediction results were better than results produced using non-parametric statistical benchmark methods. Another contribution also to corn yield prediction is the development of 4 BP models using topographic features, vegetation indices and textural indices. When compared to multiple regression method (MLR) used as a benchmark, the BP outperform the MLR in terms of prediction accuracy [8]. Another application of BP in crop yield prediction is to predict wheat yield using climatic observation data [1]. The yield was predicted with a maximum error of 45-60 kg/ha two months before crop ripening.

The objective of this study is to predict rice yield based on diseases, pests and weeds as parameters using the BP algorithm. The contributions made are, first is to transform the raw data in a suitable format required by the BP. Second is the utilization of improved unit range technique to normalize the data. Finally are the suitable parameters values for the BP obtained based on experiments using real data.

Section 2.0 describes the BP algorithm. Section 3.0 describes the methodology. Results and discussion are presented section 4.0. The paper ends with a conclusion in Section 5.0.

2.0 BP Algorithm

The BP algorithm minimizes sum of squared error E_s, measured at the output layer as defined in Equation 1 below.

$$E_s = \frac{1}{2}\sum_{j=1}^{M} d_j - y_j)^2 \qquad (1)$$

where M is number of output nodes,
d_j is the desired output, y_j is the actual output a the j-th output unit. Weight change in input and hidden layers is guided by gradient descent algorithm and represented by the relation depicted in Equation 2.

$$\Delta\omega\infty - \frac{\partial E_s}{\partial \omega} \qquad (2)$$

3.0 Methodology

The samples are obtained from Muda Agricultural Development Authority (MADA), Kedah, Malaysia ranging from 1995 to 2001. There are 4 areas with 27 locations. With 2 planting season for each year, a total of 14 seasons is generated represented by S1 to S14. There are 3 parameters affecting the rice yield namely; diseases, pests and weeds. As for diseases and pests, each consists of 12 types. However as for pests there are 11 kinds, making a total of 35 input parameters. We reduced the parameters into 3 by taking the sum of effects produced by different kinds of diseases, pests and weeds. Hence the final table consists of 3 columns representing factors affecting yield and the last column is the yield obtained. An example of the sample is depicted in Table 1.

Table 1: Effects of Parameters on Rice Yield for Locality A

Seasons	*Pests*	*Diseases*	*Weeds*	*Yield*
1/95 (S1)	63.63	189.11	91.85	3530
2/95 (S2)	73.48	283.49	120.72	4601
1/96 (S3)	139.93	148.76	72.86	3732
2/96 (S4)	36.6	331.93	53.45	3766
1/97 (S5)	54.07	25.57	122.75	2764
2/97 (S6)	74.7	21.8	161.34	3817
1/98 (S7)	101.47	55.02	93.6	3760
2/98 (S8)	125.53	229.23	136.16	4267
1/99 (S9)	285.11	96.42	307.18	4113
2/99 (S10)	127.81	71.76	161.66	4110
1/00 (S11)	165.96	28.61	232.77	4193
2/00 (S12)	101.47	114.08	176.1	3618
1/01 (S13)	184.86	41.89	323.79	4117
2/01 (S14)	88.93	73.59	428.19	3650

As follows are the steps taken to train and test the samples to predict the rice yield.

i. Data Preprocessing
ii. Determination of BP Parameters
- Determination of learning rate
- Determination of momentum
- Determination of number of nodes in the hidden layer

iii. Train the BP
iv. Test the BP

i. Data Preprocessing

Since the BP is a data-driven model that abides the "garbage in – garbage out" principle, data of insufficient quality have a tendency to brought failure to the application. BP is only as good as the input data used to train it, thus preprocessing is of paramount importance especially when analyzing real-life data to deliver a successful application [10]. In this case, the improved unit range technique is chosen as the normalization technique to normalize the data into (0.1-0.9) range shown in Equation 3[6].

$$x' = 0.9 * \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}} \right) + 0.1 \qquad (3)$$

where

x' = normalized data
x = raw data
x_{max} = a maximum data value
x_{min} = a minimum data value

Data are normalized into (0.1-0.9) range since the activation function for the backpropagation algorithm used in this study is of unipolar sigmoid, signified by Equation 4.

$$y_i = \frac{1}{1 + e^{-\lambda x_i}} \qquad (4)$$

where

y_i = the output
λ = a constant chosen to be 1, indicating the slope of the gradient descent
x_i = the input parameter

ii. Determination of BP Parameters

The values for BP parameters such as learning rate (α), momentum (β) values and number of hidden layers are problem dependent [7]. Thus in study, the values are determined empirically as reported in section 4.0.

iii. Train the BP

For each locality, data for 10 seasons are utilized for training utilizing the parameters obtained in step ii.

iv. Test the BP

Once it is found that the BP has learned what it has been taught by obtaining a minimum MSE error, weights are saved. The predictability ability of the BP is then tested on the remaining 4 seasons using the saved weights during training.

4.0 Results and Discussion

When data in Table 1 is normalized using the formula given in Equation 3, the following output is obtained as shown in Table 2.

Table 2: Normalized Data

Seasons	*Pests*	*Diseases*	*Weeds*	*Yield*
1/95 (S1)	0.108768	0.539483	0.102471	0.416984
2/95 (S2)	0.148404	0.843807	0.179511	0.9
1/96 (S3)	0.415798	0.409377	0.051796	0.526946
2/96 (S4)	0.1	0.9	0.9	0.545455
1/97 (S5)	0.070299	0.012156	0.184928	0.1
2/97 (S6)	0.153314	0.1	0.287906	0.573217
1/98 (S7)	0.261036	0.107116	0.107141	0.542188
2/98 (S8)	0.357853	0.668849	0.220713	0.818182
1/99 (S9)	0.9	0.240609	0.677083	0.734349
2/99 (S10)	0.367027	0.161094	0.28876	0.732716
1/00 (S11)	0.520542	0.021959	0.478518	0.777899
2/00 (S12)	0.261036	0.297553	0.327294	0.464888
1/01 (S13)	0.596596	0.064779	0.721407	0.736527
2/01 (S14)	0.210575	0.166994	0.9	0.482308

10 samples are then used to train the BP with different α and β values. The deviation is used as a metric to determine the suitable α and β values.

$$deviation = \frac{\sum_{t=1}^{10} out_n - out_t}{10} \quad \text{for } out_n > out_t \qquad (5)$$

$$deviation = \frac{\sum_{t=1}^{10} out_t - out_n}{10} \quad \text{for } out_t > out_n \qquad (6)$$

where

out_n - the network output
out_t - the target output

The resulting deviation for different learning rate (α) and momentum (β) values are shown in Table 3.

Table 3: Deviation for various α and β values

β \ α	*0.1*	*0.2*	*0.3*	*0.4*
0.6	0.110107	0.109659	0.109186	0.108782
0.7	0.10906	0.11114	0.110805	0.110393
0.8	0.108046	0.107499	0.106927	0.110344
0.9	0.107086	0.109286	0.108862	0.109484

From Table 3, it is observed that the combination that produces the lowest deviation is when $\alpha = 0.3$ and $\beta = 0.8$. The corresponding α and β values are then used in training to determine the optimal number of nodes in the hidden layer.

Similarly, a deviation is computed using Equation 5 and 6 in the determination of optimal number of nodes in the hidden layer, producing results shown in Table 4.

Table 4: Deviations for Different Number of Nodes in Hidden layer

Architecture	*Deviation*
3-2-1	0.037355
3-3-1	0.031384
3-4-1	0.03205
3-5-1	0.031011
3-6-1	0.032162

From Table 4, the optimal number of nodes in the hidden layer is 5, since it produces the lowest deviation.

Table 5 illustrates the NN model parameters used for training and predicting the rice yield.

Table 5: Neural Network Parameters

Parameters	*Values*
Learning Rate	0.3
Momentum	0.8
Number Nodes in the Input Layer	3
Number of Nodes in the Hidden Layer	5
Number of Nodes in the Output Layer	1

Figure 1 shows the network output and actual output during training for Locality A1. The results are encouraging.

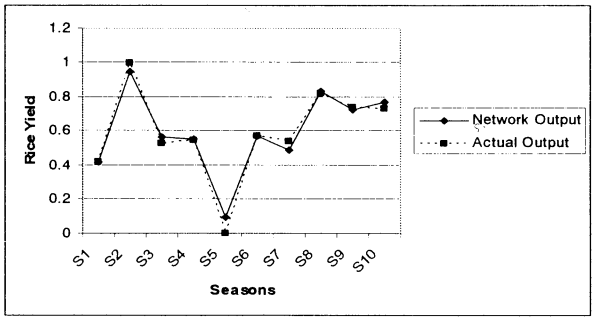


Figure 1: Graph of Rice Yield vs. Seasons for Locality A1 during training

Figure 2, illustrates the network output and actual output during prediction for same locality. Figure 3, depicts the network output and actual output during prediction for another locality C1.

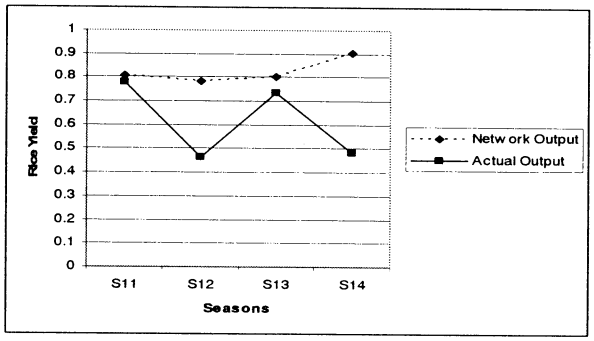


Figure 2: Graph of Rice Yield vs. Seasons for Locality A1 during prediction

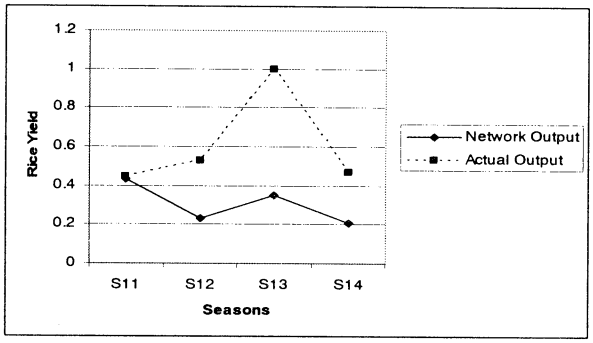


Figure 3: Graph of Rice Yield vs. Seasons for Locality C1 during prediction

5.0 Conclusion

The result of this study shows by using the improved unit range normalization technique BP is able to predict the rice yield to a deviation of less than 0.03 during training and a deviation of 0.2 during prediction. The discrepancy is due to insufficient data used for training. Although the deviation is higher during prediction, however, the trend exists based on Figure 3. The normalization technique may further be applied to normalize other types of data besides rice as long as the activation function is of type unipolar sigmoid.

References

[1] B. Safa, A. Khalili, M. Teshnehlab et al, "Prediction of Wheat Yield Using Artificial Neural Networks," in 25th Conf. on Agricultural and Forest Meteorology, 2002.

[2] J. Williams, C. Jones, J. Kiniry et al, "The EPIC crop growth model," Trans. Amer. Soc. Agric. Eng., Vol. 32, pp. 497-511, 1989.

[3] K. Sudduth, C. Fraisse, S. Drummond et al, "Integrating spatial data collection, modeling and analysis for precision agriculture," First Int. Conf. on Geospatial Information in Agriculture and Forestry, Lake Buena Vista, Florida, Vol. 2, pp. 166-173, 1988.

[4] K. Sudduth, C. Fraisse, S. Drummond et al, "Analysis of spatial factors influencing crop yield," in Proc. of 3rd Int. Conf. on Precision Agriculture, pp. 129-140, 1996.

[5] P. Atkinson and A. Tatnall, "Neural Networks in remote sensing," Int. Jour. Rem. Sen., Vol. 18(4), pp. 699-709, 1997.

[6] P. Saad, "Trademark Image Classification Approaches using Neural Network and Rough Set Theory," Ph.D. Dissertation, University Technology Malaysia 2003: in submission.

[7] S. Haykin, Neural Networks : A Comprehensive Foundation, 2nd Edn. Englewood Cliffs, NJ : Prentice-Hall, 1999.

[8] C. Z. Serele, Q. H. J. Gwyn, J. B. Boisvert et al, "Corn yield prediction with artificial neural network trained using airborne remote sensing and topographic data," in Proc. Int. Conf. Geoscience and Remote Sensing Symposium, Vol. 1, pp. 384 - 386, 2000.

[9] S. Srirengan and C. K. Looi, "On using backpropagation for prediction : an empirical study," IEEE Int. Joint Conf. on Neural Networks, Vol. 2, pp. 1284 -1289, 1991.

[10] W. A. Ahmed, "Data Preprocessing for Neural Nets," Proc. of the Int. Conf. on Artificial Intelligence in Engineering & Technology (ICAiET). pp. 683 – 686, 2002.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Study of Artificial Brain Based on Multi-Centrum Self-Coordination Mechanism

Liqun Han
School of Information Engineering
Beijing Technology & Business University
Beijing, No.11 Fucheng Rd. 100037
hlqcheng@public.fhnet.cn.net

Xuyan Tu
School of Information Engineering
University of Science & Technology Beijing
Beijing, No.30 Xueyuan Rd. 100083
tuxuyan@vip.sina.com

Abstract

Based on study of the architecture and functions of high-level nerve center system, the conception of Multi-Centrum Self-Coordination Artificial Brain (MCSCAB) and the architecture of integrated and distributed Multi-center System are presented. The mechanism of MCSCAB is discussed, and the digital and analogue integrated information patterns of MCSCAB is described.

Key works : Artificial Brain; Multi-Centrum; Self-Coordination; Artificial Intelligence

1. Introduction

Since 1940's of 20 century, to the research of brain pattern or artificial brain, people have made a lot of exploration in the fields of bionics, artificial intelligence, artificial neural network, pattern recognition, super computer and so on, and achieved a series of research fruits. For instance, from perceptron, association machine, cognition machine to cell automaton etc., they are all some kind of simplified and partial artificial brain pattern. However, since human's brain is a high complicated and subtle intelligent system which has undergone biologic evolve of long duration, on one hand, study of simulating human's brain pattern or artificial brain is an extremely difficult subject, which require long-term hard work, trek campaign and cooperation of experts from different fields; on the other hand, the system structure and coordination mechanism of the information processing of human brain provide biology prototype and innovation idea[1] for the general design and realization of artificial brain.

On the basis of study of system structure and function of human brain's high-level nerve center system, the concept of Multi-centrum Self-coordination Artificial Brain was provided and its system structure, coordination mechanism and information pattern were studied in this paper.

2. System Structure of Multi-centrum Self-coordination Artificial Brain

From the point of view of control system and information system, human brain is a large biologic control system and an information processing system which intelligence level is the highest in various biology brains, function is perfect and structure is the most complicated. Therefore, according to the study results of modern science and neurophysiology, the structure analysis method of large system cybernetics[3] is applied to analyze the architecture of human brain, taking them as the simulating object and biologic prototype of artificial brain to build artificial brain architecture pattern which simulates human brain systems' structure. The simplified architecture of multi-center self-coordination artificial brain pattern mentioned in this paper is shown as fig. 1. Each function of nerve center is described as follows.

2.1 Thinking Center

TC is the thinking nerve center pattern which simulates human cerebra. Cerebra includes pallium and fundus, and is divided into two hemisheres, known as "left cerebra" and "right cerebra". The two hemisheres are connected by callosum, constituting "thinking center" and being the highest nerve center system, which perform the functions such as thinking, volition producing, behavior controlling and movement coordinating. Left hemishere is in charge of logical thinking and right hemishere is in charge of thinking in image. They two cooperate in working and run in parallel connection. The callosum connected with both left and right hemisheres is the combine part of logos and affective, and the blend part of visual thinking and logical thinking, so it is likely that afflatus thinking emerged from here by some possession. TC bears on thinking intelligence of reasoning, association, memory, learning, decision-making, estimation and so on. Besides, TC also coordinates and controls perception center, life center and behavior center.

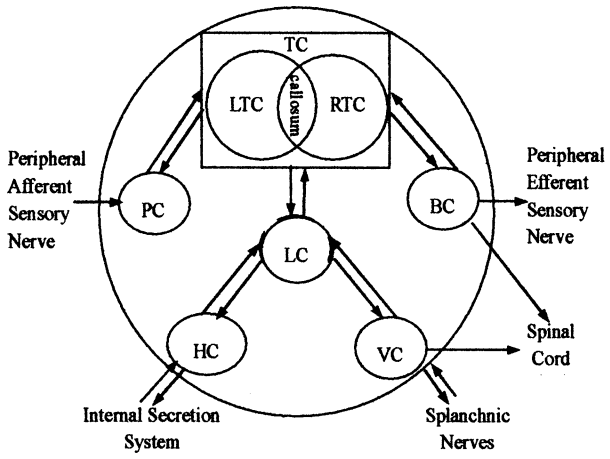


Fig. 1 A Simplified Architecture of Multi-center Self-coordination Artificial Brain

2.2 Perception Center

PC is the perception nerve center pattern which simulates human thalamencephalon. Thalamencephalon is an information center which processes various perception signals of vision, hearing, touch, smell, taste and so on. It performs time and space integration and information crasis of various multimedia and multi-pattern perception signals from peripheral afferent nerve system. PC is of perception intelligence of apperceiving, perceiving, recognizing and understanding, apperceiving outside world through peripheral nerve system.

2.3 Behavior Center

BC is the movement nerve center or behavior nerve center pattern which simulates human cerebella. Its main function is to coordinate body's movement and behavior, control body's action and carriage, keep stabilization and balance. Through low-level nerve center system and periphery nerve system, BC regulates and controls the systemic movement and carriage of body. BC is of behavior intelligence of coordinating, balance, planning, optimizing, attempering and managing the movement posture and behavior of human body, and acts on outsid world through periphery efferent nerve system.

2.4 Life Center

LC is the life nerve center pattern which simulates human brain stem constituted by midbrain, pons and medulla oblongata. The main function is to adjust and control some important physiological parameters such as heart rate, pulse, breath, blood pressure, temperature and so on, to keep steady environment physiological state of human body, in order to control human body's normal life function. Besides, LC also performs the function of coordinating and controlling hormone nerve center and viscera nerve center VC.

2.5 Hormone Center

HC is the hormone nerve center pattern which simulates human's hypothalamus-pituitary. Its main function is to emit hormone, control incretion system, adjust exudation level of various incretion hormone such as thyroxin, adrenalin, insulin, prostaglandin, sex hormone and so on. Through the circulation of body fluid such as blood and lymph, LC acts on the corresponding hormone receptor such as target cell, target organ ,etc., to adjust and control the state of human body's physiological mechanism. Therefore, HC is of the function of adjusting and controlling human body's various incretion hormone and body fluid through incretion system.

2.6 Viscera Center

VC is the viscera nerve center pattern which simulates human's medulla oblongata. Through low-level nerve center system (spinal cord) and peripheral nervous system, medulla oblongata adjusts and controls various viscera to keep the inner environment of human body steady and physiological state normal and control human body's life function. Through viscera nerve system, VC is of the function of adjusting and controlling various viscera such as heart, liver, spleen, stomach, kidney , etc..

"Multi-center Self-coordination Artificial Brain" pattern simulates human brain's high-level nerve center system in system structure. The characteristics of the structure are as follow:

(1) MCSCAB has the distributed structure of manifold information processing centers (six nerve centers named as thinking, perception, behavior, life, viscera, hormone).

(2) MCSCAB has the parallel structure which coordinates the two hemisheres in concurrent work .

(3) MCSCAB has the step-up hierarchic structure. The thinking center which controls perception, behavior and life centers is in upper level, and the hormone center which controls viscera and life centers is in lower level, so that to form a biologic system with the power of intelligent control and information processing and the characteristics of step-up hierarchic and division function.

Besides, it is the characters of MCSCAB's function that thinking center, perception center and behavior center mainly act as intelligence center to imitate human's thinking intelligence, perception intelligence and behavior intelligence separately, while life center, hormone center and viscera center mainly act as coordinating and controlling center to regulate and control the normal state of various parameters and the regular work of "viscera organs " in artificial life system.

3. Coordinating Mechanism of Multi-center Self-coordination Artificial Brain

MCSCAB simulates human's high nerve center system in system structure and its self-coordination mechanism [1~3] in function mechanism. The coordinating function of each part is described as follows:

3.1 Cerebra's Overall Situation Coordination

As the thinking center of high-level nerve center system, cerebra is of the function of coordinating and controlling overall situation. For example, pallium's whole body oriented reflect coordination, right and left hemisherers' cross parallel coordination, nerve and body fluid's dual system coordination, and the coordination of body's random movement and intent movement.

3.2 Thalamencephalon's Perception Coordination

Perception center simulates thalamencephalon's perception coordination function and performs time and space integration, information crasis and coordination to multi-patterns and mult-imedia percept information from peripheral afferent nerve system, such as vision, hearing, touch, smell, taste, sense of pain, sense of warm, etc..

3.3 Cerebella's behavior Coordination

As the movement center of MCSCAB, cerebella has the function of coordinating and controlling human's posture and movement. According to cerebra's instruction of movement or intent behavior as well as thalamencephalon's percept information about human body and outside environment, cerebella coordinates and controls human body's movement and its posture through low-level nerve center system (spinal cord) and periphery motor nerve system. Behavior coordination can keep human body (or system) balance movement, elegant posture, and harmonious behavior.

3.4 Brain Stem's Physiological Coordination

Life center simulates physiological state and coordination function of brain stem and coordinate human body's physiological and life function through coordinating and controlling viscera center and hormone center. The coordination performs in bi-direction to adjust the balance various function, for instance, the ascent and descent of blood pressure, the producing and dispelling of temperature, the inspiration and exhalation of breathe, the tone up and slow down of heart rate.

3.5 Pituitary Gland's Hormone Coordination and Controlling

Hormone center consists of hypothalamus-pituitary and is also known as "body fluid center". It has the function of coordination and di-direction adjusting to the homeostasis of various incretion in body fluid circulation system. Pituitary gland accepts di-direction adjusting of release element and restraining element excreted from hypothalamus to keep various pituitary hormone excreted from pituitary gland homeostasis. Through excreting corresponding pituitary hormone, pituitary gland adjusts and controls the corresponding exudation level of endocrine gland, so that to satisfy the needs of human body's normal physiological state.

3.6 Medulla oblongata's Viscera Coordination

Viscera center simulates viscera coordination function of medulla oblongata. It coordinates and controls dynamically various viscera through the bidirectional regulation of sympathetic and parasysmpathetic nerve of splanchnic nerve system. Meanwhile, through multi-segment and subarea coordination and control of low-level nerve center system, it controls the dilation and contraction movements of corresponding parts of body, such as thorax, abdominal muscle, etc..

To sum up, Multi-center Self-coordination Artificial Brain pattern is of characteristics as follows in function mechanism:

(1)Multilevel coordination such as thinking center to perception center , behavior center and life center; life center to viscera center and hormone center.

(2)Bidirectional coordination such as sympathetic nerve and parasympathetic nerve, ascending and descending of blood pressure, producing and dispelling temperature.

(3)Dual systemic coordination such as the nerve system's quick and subarea controlling contrasts with the body fluid system's slow and dividing controlling.

4. Information Model of Multi-center Self-coordination Artificial Brain

Similar to human brain, MCSCAB is of multi patterns in the process of acquisition, transmission and processing of information [1-3].

4.1 "Integrated-Distributed" Combined Information Pattern

Scinse MCSCAB is of the "integrated-distributed" mechanism, the acquisition, transmission and processing of the information process also bears the "integrated-distributed" information pattern. As MCSCAB is the information process center of the system, various distributed perceiving information from periphery perception nerve are centralized to high-level (or low-level) nerve center to be processed, weamwhile, various distributed instruction information from periphery motor

nerve are out of the information processing results of high-level (or low-level) nerve center.

4.2 "Serial-Parallel" Compatible Information Pattern

The information processes of acquisition, transmission and processing of MCSCAB are of "serial-parallel" compatible pattern. In each information processing center --thinking center, perception center, behavior center, hormone center and viscera center, the information processes are dividing, parallel and relatively independent., cooperate with one another under the overall situation coordinating and controlling of thinking center, and work in series. Therefore, MCSCAB is of multi-center and "serial-parallel" compatible information pattern.

4.3 "Digital-Analog" Blended Information Model

The biologic neural network of human brain is formed by mutual connection of neurons through nerve fibers. In the nerve fiber, the carrier of information transfers is nerve impulses, a kind of undamped and coded digital signal. Synapse realizes D/A transformation through the following processes: nerve impulses spur the emission of neurohumor, which in turn brings about corresponding changes at theca electricity location of nerve cell in electricity chemistry form, and then transforms itself into electricity location signal of simulation form. MCSCAB simulating human brain can take the "digit-simulation" mixture information model of human brain processing information system to buildup information processing ability.

To sum up, MCSCAB is of multiple information patterns — "digit-analog" blended, "serial-parallel" compatible and "integrated-distributed" combined, and has the function of acquisition, transmission and processing of the multi-patterns and multimedia information.

5.Summary

According to the research results of modern brain science and nerve physiology, from the point of view of bio-cybernetics and large system cybernetics, the structure analysis method of large system cybernetics is used to analyze human brain structure systematically. Making human brain structure as simulating object of artificial brain and as biologic prototype, the predigested model of "Multi-centrum Self-coordination Artificial Brain" simulating human brain in system structure is constructed.

MCSCAB model simulates high-level nerve center system in system structure and its self-coordination mechanism [1~3] in function mechanism. Similar to human brain, MCSCAB is of "digit-analog" blended, "serial-parallel" compatible and "integrated-distributed" combined multiple information patterns and has the function of acquisition, transmission and processing of the multi-patterns and multimedia information.

References

[1]Liqun Han, Theory, Design and Application of Artificial Neural Network — Artificial Neuron, Artificial Neural Network and Artificial Neural System, Chemistry Industry Press of China, 2001

[2]XunYan Tu, Bio-Cybernetics, Science Press of China, 1980

[3]XiongLi Yang, Progress in Brain Science, Science and Technology Education Press of Shanghai, 1998

[4] XunYan Tu, Large Systems Cybernetics, National Defence Industry Press, 1994

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Dynamic Air Traffic Management Using Distributed Brain Concept

Peter Sapaty [1] Vitaliy Klimenko[1] Masanori Sugisaka [2]

[1] Institute of Mathematical Machines and Systems
National Academy of Sciences
Glushkova Ave 42, 03187 Kiev, Ukraine
+380-44-2665023, +380-44-2666457 (Fax)
{sapaty, klimenko} @immsp.kiev.ua

[2] Department of Electrical and Electronic Engineering
Oita University, 700 Oaza Dannoharu, 870-1192 Japan
+81-97-554-7831, +81-97-554-7841 (Fax)
msugi@cc.oita-u.ac.jp

Abstract. Using higher-level distributed processing and control technology known as WAVE-WP (or World Processing) allows us to obtain optimized seamless spatial solutions in open dynamic environments without any central resources. Within this paradigm, the traditional message passing and communicating agents appear only on the implementation layer, drastically simplifying programming of distributed knowledge processing and control problems.

One of the strictest applications of WAVE-WP may be air traffic management in crisis situations, where a priori traffic schedules are usually useless, radar stations and control centers may become dysfunctional, and a quick, runtime, recovery and reintegration into the functional whole may be required, with dynamic routing and collision avoidance. A broader usage of the paradigm is considered too, especially for the creation of intelligent distributed robotized infrastructures for different purposes.

Keywords: WAVE-WP technology, air traffic management, distributed artificial brain, robotized infrastructures.

1 Introduction

As world dynamics is growing rapidly (military conflicts, terrorist attacks, consequences of global warming, etc), crisis management is becoming the hottest topic, with air traffic control representing a very complex problem in the crises situations.

Usually air space is divided into regions inside which traffic management is provided by local control centers with own (or shared with other centers, and located separately) radar sets, with many human controllers working cooperatively to provide smooth traffic flow in the distributed airspace, see [1,2].

Neighboring centers and radar units may communicate and handover aerial objects when the latter cross regional boundaries, but the whole picture of air traffic often remains obscure, as intensive communication between the regions is much lower and more difficult than insider them.

In crisis situations (natural disasters, wars, terrorist attacks, etc.), where a priori flight schedules may become useless, and distributed management infrastructure is damaged, much higher system vision and integration may be required to provide optimized runtime solutions covering many regions, and simultaneously for many objects in the air. Radar facilities and control centers may also get damaged, and the whole system must be dynamically updated and reconfigured to fulfill its objectives. This dynamic global vision and optimization should, however, vigorously avoid any central control, as the latter can easily become the most vulnerable part of the system.

Traditional ideologies and technologies of distributed processing and control cannot cope with these dynamic and open situations. They themselves suffer from known problems of obtaining the whole from pieces stemming from the analytical approach, being based on message passing and communicating agents, where parts are designed first and then the whole is trying to be comprehended and managed. And in crisis situations, with the necessity to survive at first place, we may need the integrity issues and controlling the whole as primary objectives.

2 Distributed artificial brain

2.1 The distributed brain concept

To attack the hot problems of management and control in dynamic and open environments, we are using the Distributed Artificial Brain concept [3] working in WAVE-WP technology, which is being developed in cooperation between the Academy of Sciences of Ukraine and Oita University in Japan.

A network of local management stations within this concept, supplied with communicating copies of the WAVE-WP interpreter (WI), converts into an integral

spatial artificial brain capable of solving any problem on distributed data without central resources. This brain interprets integral high-level spatial programs in WAVE-WP language, which provide deliberative and reactive control along with distributed knowledge processing, where global and local system scenarios, dynamically covering the network (and creating distributed infrastructures), can be launched from any unit, as shown in Fig.1.

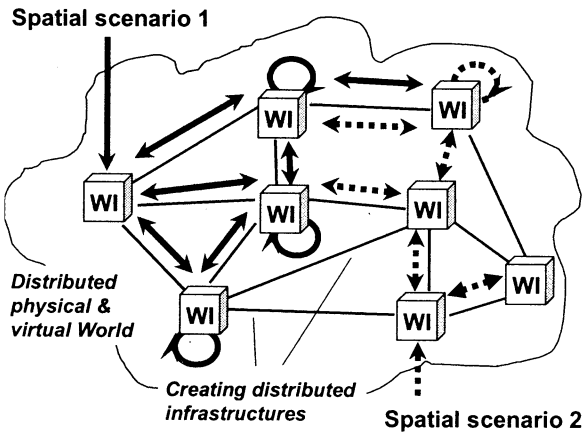


Fig. 1 Distributed artificial brain working in WAVE-WP

2.2 WAVE-WP language

The language operates in both physical and virtual distributed worlds, as well as in their integration (processing not only information but physical matter too), where any element (node or link) of the united world may have features of the both worlds. Solutions of any problems in this integrated world are represented as its parallel and coordinated navigation by some higher-level forces, or "waves" (having recursive and parallel structure), bringing local operations, control, and transitional data directly into the needed points of the worlds to perform jobs there.

WAVE-WP is not a message passing or agents-based paradigm, describing highly parallel spatial processes on a higher, semantic level, using agents or message passing on the efficient automatic implementation layer only, drastically reducing application code and complexity (up to orders of magnitude).

2.3 WAVE-WP Interpreter

General organization of the interpreter is shown in Fig.2 with processors (depicted as boxes), data structures (as ovals), and main functional and operational links between them. This (parallel) architecture can be easily implemented in both software and hardware. The communicating interpreters, if to be installed in unmanned mobile platforms (forming altogether a distributed universal robotic brain), may also be integrated with a variety of special tools and gadgets, to operate in physical world, as discussed in [3,4].

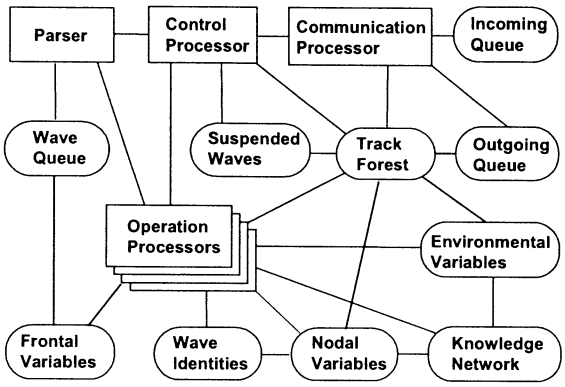


Fig. 2 General organization of the WAVE-WP interpreter.

3 Air traffic management in open and dynamic environments

We will show how the WAVE-WP technology can be used for air traffic management and control in dynamic situations -- on the example of tracing aerial objects in distributed spaces.

3.1 Creation of the neighborhood infrastructure

Radar stations and control centers do not cover the whole aerial region (say, in the scale of a country) alone, and must communicate to keep the overall observation integral and continuous. In case of indiscriminate damages, it will be very useful to provide a runtime (re)establishment of a neighborhood infrastructure, with virtual links between units reflecting the fact that their radar stations cover adjoining (generally, overlapping) regions of space, with subsequent frequent communication between the radars through this infrastructure.

We consider here a program that puts a process into each unit which regularly checks the physical distance from itself to other units, and if it is less than the sum of their radar ranges, a `neighbor` link is set up between the units. On the other hand, if the `neighbor` link already exists, but physical distance between the nodes exceeds the sum of their ranges (i.e. being mobile, the radar nodes have moved apart), such a link must be removed.

The following program, working in parallel in all units/nodes, dynamically creates and constantly updates the radar neighborhood infrastructure, (with the chosen period of 300 sec.), where the creation of new links and removal of outdated ones is allowed by only one of the adjacent nodes, to prevent unwanted competition:

```
direct#any;
repeat(
  Flocation=WHERE;
  (direct#any;ADDRESS<BACK;
    or(
      ((Flocation,WHERE)?distance<=80.0;
        or(neighbor#BACK,
            create(neighbor#BACK))),
      (neighbor#BACK;LINK=nil);done!)),
  TIME+=300)
```

An example of the created infrastructure is shown in Fig.3. The regularly updated neighborhood infrastructure helps us purify, formulate, and solve different discovery, tracking, analysis, and handover tasks in a fully distributed network mode, without central resources, using WAVE-WP.

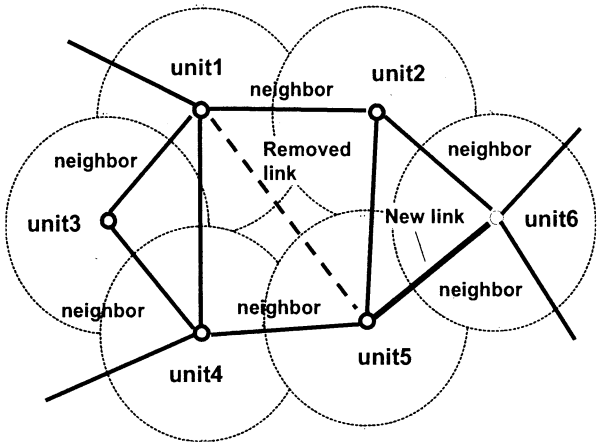


Fig.3 Dynamically created and regularly updated radar neighborhood infrastructure

3.2 Mobile tracking of aerial objects

Fully mobile character of the WAVE-WP technology allows each aerial object to be discovered and accompanied with mobile intelligence individually, while propagating via the computer network space, as shown in Fig. 4. The spatial tracking process regularly checks visibility of the object from the current radar station, as well as from neighboring stations, by launching subordinate processes in them.

If the aircraft gets closer to a neighbor, the whole tracking process, with accumulated knowledge about the object, moves to this station and continues to observe the object from there. An example of WAVE-WP code follows:

```
Fobject=TRW562;
repeat(
   repeat(Fobject~Nobjects; TIME+=10);
   any#any; Fobject~Nobjects);
USER=Fobject&&"has been lost"
```

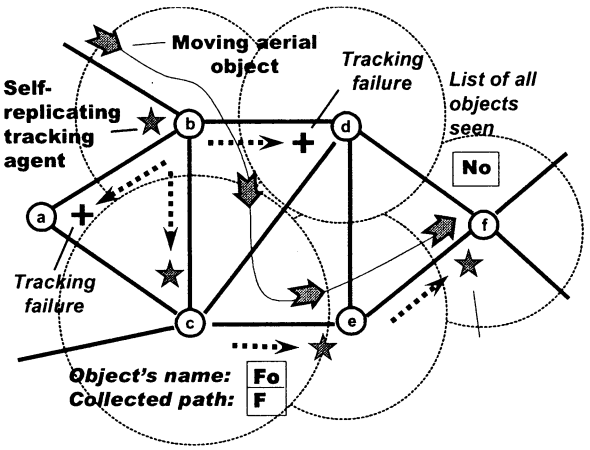


Fig. 4 Distributed networked tracking of aerial objects

where it is assumed that the controlled object has TRW562 identity, every 10 sec. its presence is checked by the current radar station, identities of all aircraft controlled from a station are kept in nodal variable `Nobjects`, and the object's identity is carried in frontal variable `Fobject`.

During this distributed tracking, the mobile intelligence may carry with it any needed payload (fully representing the individual aircraft for its distributed management).

Many aircraft can be observed and analyzed simultaneously in such a way, as shown in Fig.5. The individual tracking processes can spatially communicate with other such processes and with the global coordination processes, which can be mobile too, organizing distributed air traffic control in fully dynamic environments, and not being connected in advance to particular management centers, which can be out of operation in crises situations.

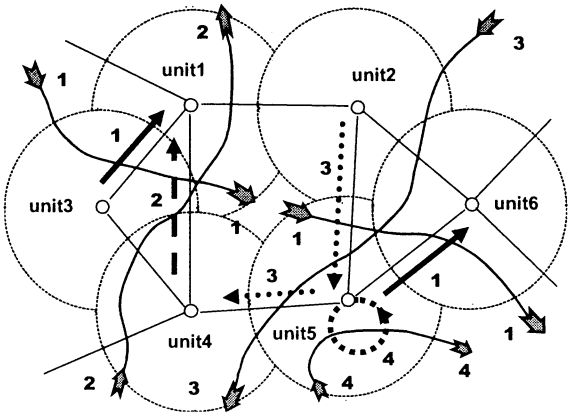


Fig. 5 Simultaneous tracking of multiple objects

3.3 Other air traffic management tasks

The WAVE-WP approach can effectively provide many other services for distributed air traffic control, for example:

- Distributed assessment and collective awareness of the existing local or global aerial situation. Intelligent distributed automated or fully automatic decisions.
- Distributed interactive simulation of the air traffic in any region, and especially in emergency situations.
- Outlining and sealing dangerous regions (say, caused by air pollution or thunderstorms).
- Distributed dynamic finding of optimal routes for aircraft, emergency rerouting, collision avoidance, etc.
- Tracing, analyzing particular air vehicles by mobile intelligence, with certain actions, and many others.

4 Other applications of WAVE-WP

WAVE-WP can also be used in a much broader scale, especially for the creation of intelligent national and international infrastructures of different natures widely using automated and fully automatic control and advanced (cooperative) robotics. A hypothetical advanced national infrastructure integrating crisis management and air traffic control centers (both manned and unmanned), with parallel spatial solutions of complex problems in WAVE-WP, is shown in Fig. 6.

Such a united system in WAVE-WP may, among many others, effectively solve the problems of distributed air defense, where multiple hostile objects penetrating the country's air (or cosmic) space can be simultaneously discovered, chased, analyzed, and destroyed, using the computerized networked radar system as a collective artificial brain operating in WAVE-WP. Many related predator-prey games had been simulated and tested in WAVE in the distributed Internet space [5,6].

Within the unified command and control infrastructures provided by the technology, different types of unmanned air vehicles may be used, for example, as possible mobile sensor, relay, or even air traffic management stations, supplementary to ground stations, especially when they get damaged or operate partially or inefficiently, in crises.

5 Conclusions

In conclusion, it can be emphasized that WAVE-WP allows for a more rational and universal integration and recovery of large complex systems than the traditional approaches, by establishing a higher level of their vision and coordination, symbolically called "over-operability", versus usual interoperability [7]. This over-operability philosophy and technology have already gone through a thorough investigation, research, and prototyping in different countries, being supported by academia, industry, and defense. It can be quickly ported to any existing software or hardware platform (or implemented in silicon).

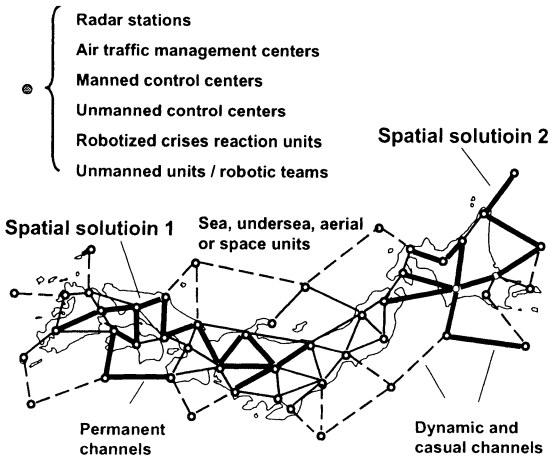


Fig. 6 An advanced national infrastructure and parallel spatial solutions in it in WAVE-WP

References

[1] Air traffic management system, Virtual Skies, http://virtualskies.arc.nasa.gov/ATM/takeControl.html

[2] Air traffic management. Revolutionary concepts that enable air traffic growth while cutting delays, http://www.boeing.com/atm/pdf/Detailed_Concepts.pdf

[3] P. Sapaty, M. Sugisaka, "Universal Distributed Brain for Mobile Multi-robot Systems", in the book "Distributed Autonomous Robotic Systems", H. Asama, T. Arai, T. Fukuda, and T. Hasegava (Eds.), Sringer-Verlag Tokyo 2002, SPIN: 10869189, pp. 434-443. Also reported at the International Symposium on Distributed Autonomous Robotic Systems DARS'02, June 25-27, 2002, Fukuoka, Japan.

[4] P. Sapaty, K. Kawamura, M. Sugisaka, R. Finkelstein, "Towards Fully Distributed Cognitive Systems", Proc. AROB 9th, Beppu, Japan, January 2004 (same proceedings).

[5] P. S. Sapaty, "Mobile Processing in Distributed and Open Environments", John Wiley & Sons, ISBN: 0471195723, New York, February 1999, 436 p.

[6] P.S. Sapaty, "Basic Distributed Control Model and Technology for Mobile Crisis Reaction Forces and Their United Air Defense, In Proc. NATO Symp. on System Concepts for Integrated Air Defense of Multinational Mobile Crisis Reaction Forces, Valencia, Spain, May 22-24, 2000.

[7] P.S. Sapaty, "Over-Operability in Distributed Simulation and Control", The MSIAC's M&S Journal Online, Winter 2002 Issue, Volume 4, No. 2, Alexandria, VA, USA, http://www.msiac.dmso.mil/journal/WI03/sap42_1.html.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Tierra Sonification: An Audio Display System for Tierra

Joseph Hart
ATR Human Information Science Laboratories
Kyoto,Japan
jhart@atr.co.jp

Rodney Berry
ATR Media Information Science Laboratories,
Kyoto,Japan
rodney@atr.co.jp

Abstract

Tools exist for the graphical display and analysis of data from Tom Ray's Tierra environment. Real time changes, however, can only be identified while the researcher is present and actively observing the display. In this paper, we propose a basic scheme for an auditory display system for Tierra that permits real time observation without monopolizing the attention of the observer. We describe our first attempts at implementation.

1. Introduction

Tom Ray's *Tierra* software is one of the earliest examples of artificial life on the computer.[1] It consists of a virtual computer in which strings of code reproduce and compete for computing resources. Since the primary activity takes place invisibly in the memory of the computer, the *Beagle* tool is used to visualize the internal activity of Tierra.

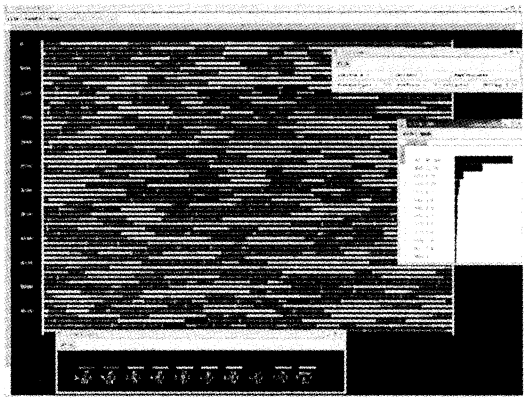


Figure 1 Typical Beagle Display

However, visually detecting significant events as they happen in real time requires someone to be watching the screen at the precise moment that the event occurs. Noticing such an event would otherwise involve carefully examining graphs and other records with the hope of recognizing some unusual pattern. For this reason we propose a basic scheme for the auditory display of aspects of Tierra's behaviour in order to augment the already existing graphical and numerical display and analysis tools.

2. Sonification or Auditory Display

The field of *auditory display* became established over a decade ago with the first International Conference on Auditory Display in Santa Fe.[2] Sound is one of the ways in which we gather crucial information about our environment, especially about things that are outside our field of view. Sound is also one of the 'vital signs' we use when discerning animate from inanimate things in our environment, so a sound-based display may help to convey a sense of the 'aliveness' of an artificial life environment. In the case of Tierra, the authors would like to know about changes in the behavior of the environment while it is running without having to constantly watch for changes in graphic or numerical information presented on a video monitor.

2.1. Pure Data

The sound generation software we chose for our first attempts at sonification is Pure Data (PD) by Miller Puckette.[3] [4] Pure data is a free software synthesis environment that allows the user to construct sound generation systems by using a mouse to connect symbols on the screen, flow-chart fashion, with graphical 'wires'. The environment currently runs under Linux, MacOS X, Windows and IRIX.

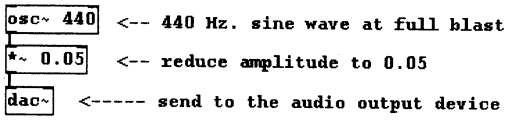


Figure 2 A simple sound generator made in PD

Apart from oscillators, PD provides a range of other sound generating units, filters and amplifiers, as well as many of the basic tools for control and data manipulation one might expect to find in a computer language. As well as communicating via MIDI signals, PD can read data from text files and communicate with other programs via network sockets. This makes it very convenient as a tool for interpreting data as sound.

2.2. Mapping to Sound

Of all the information that can be extracted from Tierra, we decided that the changing populations of various size classes would be of most interest for monitoring in real time. To Begin with, we plan to represent each size class as a distinct frequency in the audio spectrum. The population of each size class would determine the amplitude of each frequency component. However, because there are potentially several hundred thousand different size classes in a run of Tierra, we need to somehow scale this down to a hundred or so sine-wave generators.

In addition, our perception of audio frequency is not linear. Instead, we tend to hear each doubling of frequency as an octave higher than the last. For this reason, we have re-scaled the range of size classes into groups arranged across the midi note range (PD allows for fractional midi note numbers for frequencies that fall outside the normal MIDI scale).

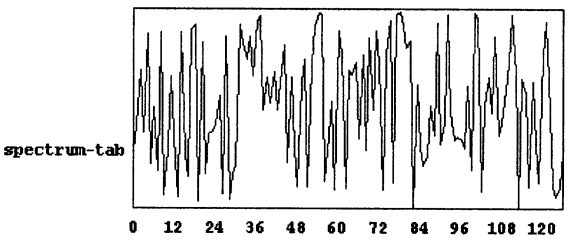


Figure 3 Spectral envelope based on size classes

2.3. Auditioning

The results are crude but promising. Tuning the sine tone generators to harmonics of a fundamental frequency created a slightly musical quality to the sound and it was possible to identify changes in the size-class populations. Mapping larger size-classes to lower pitches was intuitive since larger things typically have deeper sounds in the real world.

3. Conclusions and Future Plans

The work is still in its infancy but we have established that it is possible and worthwhile to use a tool such as PD as a form of auditory display for Tierra. It is possible to hear changes in real time while one's attention is directed elsewhere, much as we only notice the sound of the air conditioner when it turns off. In the future, we plan to experiment with monitoring other Tierra parameters in this manner.

4. Acknowledgements

The research reported here was supported in part by a contract with the Telecommunication Advancement Organization of Japan entitled, *"A Study of Innovational Interaction Media toward a Coming High Functioned Network Society" and "Research on Human Communication"*.

5. References

[1] Ray, T. S. 2001. *Overview of Tierra at ATR.* In: "Technical Information, No.15, Technologies for Software Evolutionary Systems". ATR-HIP. Kyoto, Japan.
http://www.his.atr.co.jp/~ray/tierra/

[2] Kramer, G.Ed. 1992 Auditory Display, SFI studies in the Science of Complexity, Proc Vol. XVIII Addison Wesley pp. xv, 1-77

[3] Puckette, M. 1996. Pure Data. *Proceedings, International Computer Music Conference. San Francisco*: International Computer Music Association, pp. 269-272.

[4] http://crca.ucsd.edu/~msp/Publications/icmc96.ps
http://crca.ucsd.edu/~msp/software.html

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Aspect of Non-Story Processing and Film Rhetoric Composition in the Narrative Generation Mechanism

A. Kanai
Research Fellow of Japan Society for the Promotion of Science
Yamanashi Univercity
43-21 Enokigaoka, Aoba-ku, Yokohama, Kanagawa, 227-0063, Japan

T. Ogata
Department of Medicine and Engineering
Yamanashi Univercity
4-3-11 Takeda, Kofu, Yamanashi, 400-8511, Japan

Abstract

We propose an artificial film rhetoric composition system, which can treat the aspect of non-story processing. Moving image creation is done by the sender choosing the type of rhetoric that fits his/her strategy, either story (S) or non-story (NS) type. The system is a part of the whole narrative generation mechanism, which includes aspects of both S and NS processing. Based on NS processing, the system can emphasize non-story rhetorical aspects of moving images. The aspects can cause cognitive effects that do not arise through comprehending a story, but instead, arise through the audiovisual situation itself. The cognitive effects are important in art film, music video, advertising, trailer and so on. In this paper, we make films using NS-type rhetoric, whose source moving image are made from the part of story processing in the whole narrative generation system.

1 Introduction

We define a story as all implicit or explicit events in a narrative. A moving image (e.g. a film) usually has a rhetoric that tells a story to conform to the viewer's internal constraint to the story comprehending. Kanai (1999) defined the rhetoric of the moving image [7], based on Chatman's (1990) definition [3], as a combination of moving image techniques reflecting a sender's purpose. Story-type (S-type) rhetoric, i.e., rhetoric of a moving image with the purpose of telling a consistent story, is only one possibility; there is also non-story type (NS-type) rhetoric, i.e., rhetoric whose purpose is other than telling a consistent story. NS-type rhetoric is important in advertising, art films, video art, trailer and so on, yet NS-type rhetoric has not been sufficiently studied in the field of cognitive science or artificial intelligence. In this paper, we made films using NS-type rhetoric, whose source moving images were made from the part of story processing in the whole narrative generation system.

2 Non-Story and Story Processing

We propose the film rhetoric composition system (FRCS) that is a revised version of the system proposed by Kanai, Shinoha, & Ogata (2002), and Kanai (2003) [10] [11]. The former system treated the aspect of non-story processing part of the whole film generation mechanism, which includes aspects of both story and non-story processing. The former system can emphasize the NS-type rhetorical aspects of the original film. However, the ability to combine NS- and S-type rhetorical aspects is important in film rhetoric generation mechanism. We enhanced the system so it can use complicated continuity and rational relationships based on literature theory such as Genette (1972) and film theory such as Bordwell (1985) [4] [2].

Moving image creation is done by the sender choosing the type of rhetoric that fits his/her strategy, either S- or NS-type. The FRCS can generate either rhetoric from the source video. In making the system, we focused on the continuity and discontinuity, rational and irrational relationships in the rhetoric of the moving image. The FRCS can make moving images by using one or more NS- and S-type rhetoric approaches through which the user can see all the possibilities of rhetoric creation. Continuity is used in many S-type rhetoric. For example, spatial continuity ensures that a viewer can understand the characters' relative locations even when they are not in the same frame [1]. Continuity can be created by putting identical elements in continuous shots. FRCS uses approaches for making stories (story processing) based on continuity and rational relationships by which the system

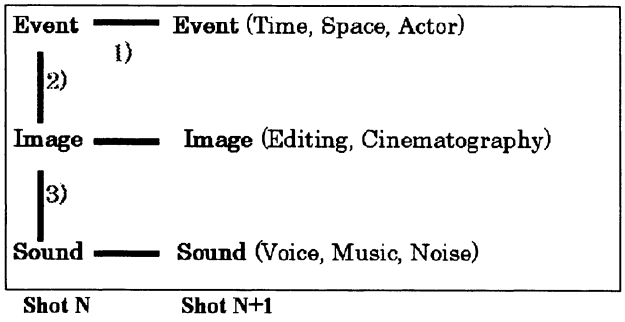


Figure 1: Approaches for making discontinuities and continuities in continuous shots

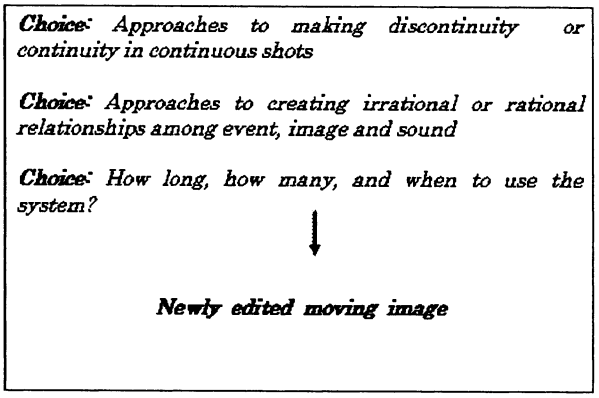


Figure 2: Outline of the FRCS

can make S-type rhetoric.

In contrast, discontinuity must be emphasized to make NS-type rhetoric. Here, based on the destabilization made by the discontinuity, the viewer sometimes cannot comprehend a consistent story from the images. Consequently, attention to the rhetoric itself is enhanced. Discontinuities can be created by placing different elements in continuous shots (non-story processing).

Discontinuities or continuities will be created based on the three relationships below, in the rhetoric of the moving image.

Approach 1) Relationship between rhetorical elements of two shots

Approach 2) Relationship between one event and image

Approach 3) Relationship between sound and image

These relationships are also illustrated in Figure 1. NS-type rhetoric may also have different kinds of continuity, such as actor, object, sound, or tone of photography. The interplay between continuity and discontinuity is important.

In NS-type rhetoric, one or more of the above discontinuity approaches may be used to cause cognitive effects through the audiovisual situation itself. On the contrary, in S-type rhetoric, rational relationships about above three approaches are important.

3 Film Rhetoric Composition System

Base on database of shots, FRCS uses the three relationships of discontinuity and continuity approaches, discussed above in Figure 1, to compose newly edited moving image. The FRCS lets the user choose the number of such discontinuities or continuities and their cycle, as well as the first frame of the moving image. It then retrieves the shots from the original moving image that fit the chosen strategy. The shots are edited as a sequence of shots and outputted (Figure 2).

The interaction between story and non-story processing is one of the most important aspects of film composition. So, FRCS also uses complicated continuity, based on Genette's theory of narratology [4]. The theory consisted of tense, mood, and voice. We especially focus on "mood." "Mood" is defined as control of narrative information such as distance, perspective and alterations. Thanks to "mood," we can focus not only character but also background situation or object of the moving image. FRCS uses alterations such as paralipsis and paralepsis to control narrative information.

For example, rhetorical elements of the moving image such as color, brightness, speed, number of the screen, frequency, and sound can be reduced or expanded from the original moving image by using the system. The former case is the paralipsis. The later case is the paralepsis. Based on consistent paralipsis or paralepsis, complicated continuity can be made in a moving image.

We also focus on Bordwell's theory of parametric narration in the fiction film [2]. Bordwell showed two approaches, ascetic and replete, to make intrinsic

norms in a film. Ascetic corresponds to paralipsis, and replete corresponds to paralepsis. Bordwell argued "ascetic" as the strategy, in which the film limits its norm to a narrower range of procedures than are codified in other extrinsic norms, and "replete" as the strategy, which can creates an inventory of paradigmatic options. In our system, the extrinsic norms depend on the original moving image.

4 Story Processing in the Narrative Generation Mechanism

Ogata (2002) proposed the narrative generation mechanism, which can treat various narrative media, such as literature, music, hyper-comic, and film [?]. The story processing part of the narrative generation mechanism uses the theory of Propp (1968), Ogata (1991), and Hosaka & Ogata (2003) to make stories [14] [12] [?]. We made films using NS-type rhetoric, whose source stories and moving images (CG Animation) were made from the part of story processing in the whole narrative generation system [5]. One source story was shown below.

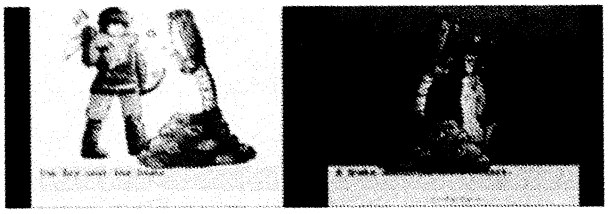
Long, long ago...
A snake abducted the princess.
A boy took a departure from the kingdom to the forest.
A Leg-less man gave the boy an apple.
...[omission]...
The boy beat the snake.
...[omission]...
The boy married with the princess.

The story was also represented with CG animation by the narrative generation system.

5 Non-Story Processing in the Narrative Generation Mechanism

We used the FRCS to make example films incorporating two discontinuity approaches (Approach 1 and 2), from the CG animation of the story made by the narrative generation mechanism. Tone of CG animation was chosen as the consistent element to establish continuity.

For example, we made one film whose shots consisted of less than five frames. A part of the scenario of newly edited CG animation including discontinuity approach is shown below.



Shot 1 *Shot2*

Figure 3: Shot 1 and shot 2

Long, long ago...
Shot 1) The boy beat the snake (Discontinuity Approach 2).
(Discontinuity Approach 1)
Shot 2) A snake abducted the princess (Discontinuity Approach 2).
...[omission]...

In the original animation film, shot 1 and shot 2 are not continuous shots and the order is shot 2 - shot 1. On the other hand, because FRCS used discontinuity approach 1, the generated film's order is changed to shot 1 - shot 2. FRCS cut a few frame from all the shots of the original film. The story of the original film was thus altered by the cuts. And we used consistent paralipsis and paralepsis. In this case, we reduced brightness and color, and expanded speed and frequency (discontinuity approach 2). Shot 1 and 2 are shown in Figure 3.

The FRCS can make various kinds of NS-type rhetoric. The user can decide which approaches to use for making discontinuities or establishing continuity. By using this computational approach, we can make many different films from the same shots or story, and study the nature of NS-type rhetoric.

6 Discussion

In this section, we discuss application of FRCS except for art film making. The system can be used in the field of advertising, video abstracting, trailer making, and so on.

For example, in advertising, making brand knowledge structures (brand processing) is important. To

do so, the advertising rhetoric must be able to "reset the viewpoint." Useful advertising is done by producing intensive cognitive effects not only by the ads story being comprehended but by the rhetoric itself such as irrational relationships existing between two events, event and image, or image and sound. The FRCS can make moving images by using one or more NS-type rhetoric approaches (discontinuity) through which the user can see all the possibilities of rhetoric creation.

To make a viewer do brand processing, not only events but also elements relating to the brand must be emphasized. For example, in a "make up foundation 1" advertisement analyzed in Kanai & Kato (2001), the character's actions do not associate with one event [9]. Space time, and action is changed every time with the shot changes. But there are associations of the brand character and the brand product. In this case, a viewer can not understand consistent story and story processing was retrained. So, attention to the character herself is enhanced and character processing was done to promote brand processing. Moreover tone of photography is changed every time from sixth to ninth shot with the shot changes in this ad. The change cause viewer do mood processing and promote brand processing. FRCS also can make a film like "make up foundation 1" advertisement.

7 Conclusions

We developed a film rhetoric composition system (FRCS) that can emphasize various rhetorical aspects of moving images. The FRCS can make various kinds of NS- and S-type rhetoric, by creating irrational or rational relationships between 1) two shots, 2) event and image, and 3) image and sound. The intensity of viewer's cognitive effects differs according to the strategy for the relationships.

Acknowledgements

This research was partially supported by the Ministry of Education, Culture, Sports, Science and Technology, Grant-in-Aid for JSPS Fellows.

References

[1] D. Bordwell, & K. Thompson, *Film Art: an Introduction, Fifth Edition*, McGraw-Hill, 1997.

[2] D. Bordwell, *Narration in the Fiction Film*, Madison, Wisconsin: The University of Wisconsin Press, 1985.

[3] S. Chatman, *Coming to Terms*, Cornell University Press, 1990.

[4] G. Genette, "Discourse du recit, essai de methode," In *Figures* III, Seuil:Paris, 1972.

[5] Y. Hosaka, & T. Ogata, "Story Generation Based on Narratology and Representation with CG Animation," *Proceedings of 17the Congress of the International Association of Empirical Aesthetics*, 561–564, 2002.

[6] Y. Hosaka, & T. Ogata, "A Study of Story Generation and Representation – A Simulation of V.Propp Theory as a Starting Point," *Proceedings of the Fourth International Conference on Cognitive Science*, 2003.

[7] A. Kanai, "Rhetoric and Story Contents in Film Cognition," *Proceedings of the Second International Conference on Cognitive Science*, 177–182, 1999.

[8] A. Kanai, "The Story as a Constraint on Film," *Proceedings of the Third International Conference on Cognitive Science*, 503–506, 2001.

[9] A. Kanai, and Y. Kato, "Effect of Advertising Rhetoric on Viewer's Cognitive Process," *Journal of Advertising Science*, **42**, 87–100, (In Japanese), 2001.

[10] A. Kanai, T. Ogata, & K. Shinohara, "Film Rhetoric Generation System for Intensive Cognitive Effects," *Proceedings of 2002 IEEE International Conference on Systems, Man and Cybernetics*, MP2H3, 2002.

[11] A. Kanai, "Non-Story Type Film Rhetoric and Cognitive Effects," *Proceedings of the Fourth International Conference on Cognitive Science*, 259–264 2003.

[12] T. Ogata, & Tarano "Explanation-Based Narrative Generation Using Semiotic Theory," *Proceedings of National Language Processing Pacific Rim Symposium 91*, 321–328, 1991.

[13] T. Ogata, "The concept of system narratology: from the viewpoint of expanded literary theory," *Proceedings of 3' d international workshop of literature in cognition and computer (iw LCC) in held PRICAI2002*, W7-1 1-10, 2002.

[14] V. Propp, *Morphology of the Folktale*, University of Texas Press, 1968.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

ATR Artificial Brain Project: Key Assumptions and Current State

Andrzej Buller, Hitoshi Hemmi, Michal Joachimczak[1], Juan Liu,
Katsunori Shimohara, and Adam Stefanski[2]
ATR International, Human Information Science Laboratories,
2-2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288 Japan
{buller; hemmi; mjoach; juanliu; katsu; ajstefan}@atr.co.jp

Abstract This paper presents key assumptions and the current state of the ATR Artificial Brain Project aimed to create Volitron—a device in which one could observe emergent thinking to be recognized from specific communication behaviors. The project consists of two complementary themes: Psychodynamic Architecture (PDA) and Brain-dedicated Evolvable Hardware (BEH). According to the theory underlying PDA, if a machine equipped with a set of counters representing certain psychic tensions and a device that forces the machine to continuously attempt to discharge the tensions, then the machine may autonomously develop its intelligence to eventually reach the "strong AI" level. For BEH we develop *q*Cellular-Automata (*q*CA), in which groups of uniform logic primitives (*q*-cells) serve as spike-train-processing units, and Pulsed Para-Neural Networks (PPNN) that can be evolved (using fuzzified signals and a genetic algorithm combined with hill climbing) and converted into *q*CA. Although, ultimately, every BEH cell is to be a physical entity, we develop two interim solutions: a Universal Spiketrain Processor (USP) and an FPGA-based *q*CA Machine. Selected PDA/BEH-related concepts were tested using a simulated robot equipped with MemeStorms—a special Working Memory in which conflicting ideas fight for access to actuators.

1 Introduction

The ATR Artificial Brain Project is being conducted in the framework the Emergent Communications Mechanisms Project aimed to create novel technologies to reach the frontiers of new possibilities in human-machine communication [20]. The final product is to be *Volitron*—a device in which one could observe the phenomenon of emerging thought to be recognized from specific communication behaviors [6]. The project consists of two complementary themes: Psychodynamic Architecture (PDA), and Brain [-dedicated] Evolvable Hardware (BEH). PDA (described in Section 2) is to contain all functional agencies and communication channels necessary to host emergent thought. It is also to facilitate orchestrated work of researchers synthesizing new agencies, inserting them into the "living" Volitron, and then testing and modifying them.

Since the famous experiments with simulated worlds populated by primitive evolving creatures gave a limited hope that a thinking machine can be obtained via simulated evolution [3], while, on the other hand, successful employments of genetic programming to solve defined modest-sized problems are reported [18], we develop the *Evolutionary Short-Cut Doctrine* according to which isolated functions are evolved separately with fitness function referring to behavior of a Volitron-driven agent, using, if necessary, a simplified simulation model of the agent and its world [8].

Currently, the Volitron is being developed using the *Platform*, i.e. a dedicated client-server system using a number of workstations. The clients can be implemented as C++ programs, as well as circuits synthesized using the *NeuroMaze*™ *3.0 Pro* [10]. But one day we will want to install Volitrons in heads of life-size robots. In this case, every cubic millimeter in a robot's head will be worth its space in gold, so general-purpose-microprocessors with gigabytes of memory occupied by unwanted library functions seem to not to be a prospective solution [8]. BEH (introduced in Section 3) is intended to use as many logic primitives as truly necessary, and, which is more important and seem to be evolvable.

2 Psychodynamic architecture (PDA)

Since the possibility to have empirical evidence supporting any theory of mind is limited, the ATR Artificial Brain Group had to choose the most suitable ideas and verify them through behaviors demonstrated by its creation. Our choice is a selection of ideas proposed, first of all, by Sigmund Freud, Jean Piaget, and Marvin Minsky. Based on the selection, a Psychodynamic Architecture (PDA) is being developed.

[1] On leave from Gdansk University of Technology, Poland
[2] On leave from Simon Fraser University, Vancouver, Canada

It is assumed that the only mission of PDA-based agent is to seek for pleasure for itself. The pleasure, as Freud proposes, is a quick discharge of a psychic tension (cf. 15:3]), which is represented in PDA as a state of a certain counter. The architecture is being considered in terms of agencies constituting a certain instance of the *Society of Mind* (see [17]). PDA, on the most general level of consideration, consists of two coupled agencies: *Tension Accumulator (TA)* and *Tension Discharger (TD)*. *TA* includes a sensory system, a bundle of input channels, and a collection of tension counters. *TD* includes: a set of actuators, a bundle of output channels, a set of blocks responsible for tension-discharging behaviors, and a Working Memory regulating the timing of discharge of particular tensions. Some groups of tension counters and behavior blocks constitute sub-agencies facilitating action planning, long-term memories, and defense mechanisms regulating psychic balance (Figure 1).

Although an ability to imitate other agent's behaviors is currently one of the most essential factors to qualifying a machine as an intelligent entity, the point is to build imitating systems without modules dedicated to imitate defined things [14]. Therefore, we develop a tension-based imitation. Volitron is to have an agency that, based on the Model of Perceived Reality, creates a Model of Desired Reality (e.g. an image of the agent behaving the same way as another, already perceived agent), then generates a tension (dependent on the difference between the created and perceived model), and rewards or punishes the agency responsible for discharging of this tension (cf. [13])

Freudian psychology does not provide any idea about the essence of psychic forces and mechanisms of their interactions. We propose to represent the forces as populations of identical pieces of information representing concepts of interest. We call them *memes* [11][7]. A working memory, called *MemeStorms*™, is a theater in which populations of memes fight for access to actuators. We have performed a number of simulation experiments with such kind of memory [12][7]. The results were comparable with psychological evidence about intrinsic dynamics of judgment provided by Andrzej Nowak and Robin Vallacher [19].

The question that currently seems to divide the AI community the most is whether intelligent minds use a *world model* or not. While one camp recognizes a world model as an indispensable part of intelligent systems [1], the idea of intelligence without representation is being promoted by the opposite camp [2]. Rodney Brooks and his coworkers collected psychological evidence supporting the statements that humans do not build an internal model of the entire visible scene and that there are multiple internal representations that are not mutually consistent [4]. PDA offers a compromised solution. Volitron does not and cannot build an internal model of the *entire* scene because the aforementioned *MemeStorms*™ employs volatile swarms of memes representing only selected parts of the scene and those obtained on request from long-term memories that represent only the most necessary elements of perceived reality [11]. The Model of Perceived Reality is being updated for the agent's entire life and never becomes perfect. It is also never accessible as a whole because of the limited capacity of Working Memory. A Piagetian way of mind development [21] is being modeled toward machine long-term memory self-improvement. Another argument for representations and world models is the possibility to improve pattern recognition via a mechanism proposed by Jerome Brunner, i.e. comparing a new image with previous recognitions [5].

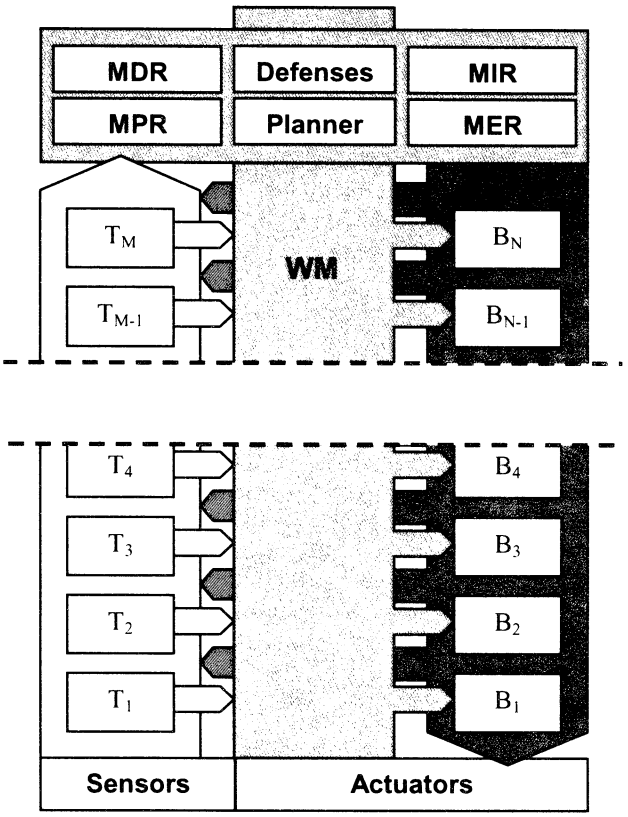


Figure 1. Outline of Psychodynamic Architecture (PDA). The column based on Sensors serves as tension accumulator, while the column based on Actuators serves as tension discharger. MPR – Model of Perceived Reality, MDR – Model of Desired Reality, MER – Model of Estimated Reality, MIR – Model of Ideal Reality, WM – Working Memory, T_1, T_2, ..., T_M – tensions, B_1, B_2, ..., B_N – behaviors. The tensions continuously grow, while the behaviors are continuously used and modified or replaced to efficiently discharge the tensions. The Planner and the models support discharge-oriented activities. In case of a deadlock, the Defenses can displace the tensions or tension-making memories. The agent's only mission is to have a possibly rich record of successful tension discharges.

We simulated a robot equipped with a limited PDA-based controller that demonstrates several life-like

behaviors, including excitement caused by a toy and learning how to cope with hunger (battery low), or with boredom (lack of object of interest). One of recent demonstrations concerns perceptual uncertainty and conflicting tensions. When an object belonging to the category 'exciting objects' were distant or partially visible, the robot chased it "less willingly" or 'hesitated' whether or not to chase it, depending on the certainty that this is really the object to be chased. Hesitations appeared within a narrow range of the level uncertainty, while the limits of the range were modified by the hunger-related tension.

3 Brain-dedicated hardware

The psychodynamic functions can be coded in several ways. We develop Pulsed Para-Neural Networks (PPNN) that are graphs consisting of processing nodes and directed edges, called axons [9]. The nodes represent delayed functions operating on spike-trains, where all spikes have the same amplitude. Axons represent pure delays. Although several node functions have been investigated, in this paper we discuss PPNNs in which both nodes and axons receive/return spikes at discrete moments of time called clocks, each node returns a pulse at clock t if it received one and only one pulse at clock t-1. PPNN graphical notation uses squares representing nodes, tiny arrows representing delays equal to 0, and thick labeled arrows representing non-zero delays, where a given label appears as a particular number of clocks or a symbol of a variable (Figure 2). Since a node with a constant input equal to 1 can serve as delayed logical NOT function, while three appropriately connected nodes may constitute a delayed logical AND function (Figure 3), PPNN appears to be universally complete, which means that for any desired manipulation on spiketrains there is an appropriate PPNN. We investigate various methods of evolving PPNNs (see Figure 4).

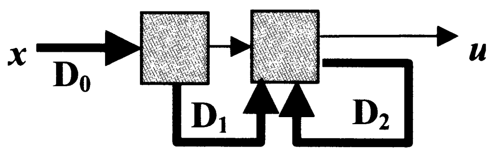


Figure 2. PPNN-based timer. For $\boldsymbol{x}$ = 1 0 0 0... and D_0, D_1 and D2 equal to 1, 40, and 7, respectively, $\boldsymbol{u}$ = 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0... In the PPNN schemes every non-labelled arrow means a connection of no delay.

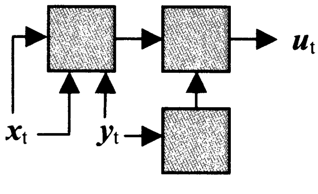


Figure 3. PPNN serving as a delayed AND (adopted from [*]). $u_{t+2} = x_t\, y_t$ (adopted from [15]).

The currently investigated concept of BEH is considered as a battery of 2D qCellular-Automatic (qCA) panels, where states of uniform q-cells represent both a given PPNN and layout of pulses (Figure 5). Any PPNN can be easily converted into an equivalent qCA. Although, ultimately, every BEH cell is to be a physical entity (see Figure 6), we develop two interim solutions: a Universal Spiketrain Processor (USP) and a qCA Machine (qCAM). USP is a dedicated digital device that directly emulates a given PPNN using a pointer moving along a FIFO buffer representing each delaying edge. The heart of the qCA Machine is an FPGA circuit that emulates a 2-dimensional array of q-cells and multiplexes input/output signals.

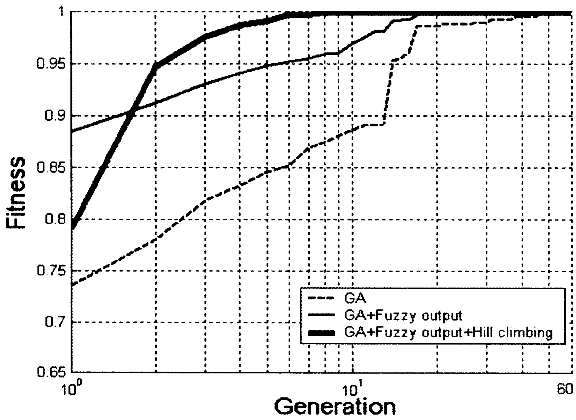


Figure 4. Evolving PPNN. The presented example concerns evolutionary search for two unknown delays in a scheme built on two parallel timers. Fuzzifying output signals and combining Genetic Algorithm with a hill-climbing (possible owing to the fuzzification) accelerates the evolution. Evolved PPNN can be easily converted into equivalent qCellular Automata.

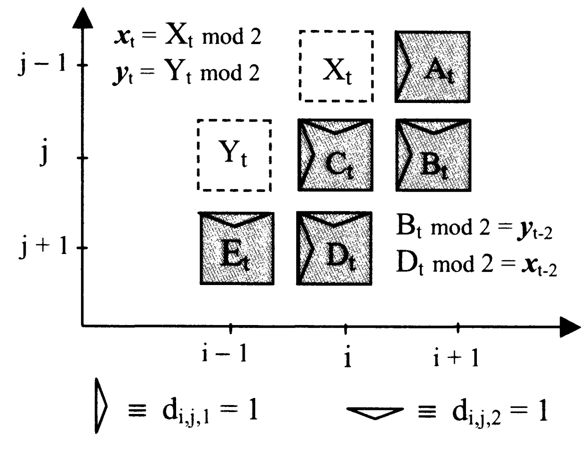


Figure 5. Example of qCellular Automaton (qCA). Every q-cell has its location i,j and a state defined by an integer Z_t defined by $d_{i,j,4}d_{i,j,3}d_{i,j,2}d_{i,j,1}d_{i,j,0}$ (a series of binary values) where $d_{i,j,0}$ serves as a pulse to be propagated. The group A, B, ..., E of q-cells serves as a flat overpass (worked out based on [15]).

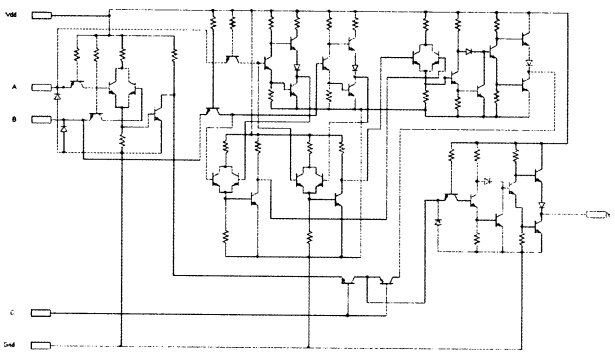


Figure 6. Transistor-based model of q-cell. The idea of millions of such cells pasted into a surface is a candidate vision of an ultimate hardware for artificial brains.

4 Concluding remarks

The theory underlying PDA proposes that if a machine were equipped with a set of counters representing certain "psychic tensions" and a device that forces the machine to search for the ways how to discharge the tensions, then the machine might autonomously develop its intelligence to eventually reach the level of "strong AI".

The Universal Spiketrain Processor (USP) and qCA Machine are complementary tools for BEH-oriented research and are to replace the obsolete CAM-Brain Machine (CBM). The stations constituting the Platform will be gradually replaced with clusters of USP or qCAM units to by one day replaced with an ultimate synthetic neural tissue.

Acknowledgement: This research was conducted as a part of the *Research on Human Communication* supported by the Telecommunications Advancement Organization of Japan (TAO). A. Stefanski's work was made possible owing to the Co-op Japan Program.

References

[1] Albus J.S. & Meystel A.M (2001) *Engineering of Mind: An Introduction to the Science of Intelligent Systems*, J.Wiley & Sons.

[2] Brooks R.A. (1991) Intelligence Without Representation, *Artificicial Intelligence Journal, vol. 47*, 139-160.

[3] Brooks R. (2002) *Flesh and Machine*, Pantheon Books.

[4] Brooks R.A., Breazeal (Ferrel) C., Irie R., Kemp C.C., Marianović M., Scassellati B. & Williamson MM (1998) Alternative Essences of Intelligence, *Proceedings, AAAI-98, July 26-30, Madison, Wisconsin*, 961-968.

[5] Bruner J.S. & Anglin J.M. (1973) *Beyond the Information Given: Studies in the Psychology of Knowing*, W.W.Norton & Co.

[6] Buller A (2002) Volitron: On a Psychodynamic Robot and Its Four Realities, In: Prince CG, Demiris Y, Maron Y, Kozima H, Balkenius C (Eds.) *Proceedings of the 2nd International Workshop on Epigenetic Robotics: August 10-11, Edinburgh, Scotland, Lund University Cognitive Studies 94*, 17-20.

[7] Buller A. (2002) Dynamic Fuzziness, *Proceedings, 7th Pacific Rim Conference on Artificial Intelligence, Tokyo, Japan, August18-22*, 90-96.

[8] Buller A (2002) In Quest of an Artificial Brain, *Proceedings of the Fifth International Conference on Human and Computer (HC-2002), September 11-14, Aizu, Japan*, 195-201.

[9] Buller A. (2002) CAM-Brain Machines and Pulsed Para-Neural Networks: Toward a hardware for future robotic on-board brains, *The Eight International Symposium on Artificial Life and Robotics (AROB 8th '03), January 24-26, Beppu, Japan*, 490-493.

[10] Buller A., Joachimczak M., Liu J. & Stefanski A. (2003) NeuroMazes: 3-Dimensional Spike-train Processors, *2nd WSEAS International Conference on Electronics, Control and Signal Processing (ICECS '03), December 7-9, Singapore.*

[11] Buller A. & Shimohara K. (1999) Decision Making as a Debate in the Society of Memes in a Neural Working Memory, *The J. of 3D Images, 13* (3), 77-82.

[12] Buller A.. & Shimohara K.. (2000) Does the 'Butterfly Effect' Take Place in Human Working Memory? *The Fifth International Symposium on Artificial Life and Robotics (AROB 5th '00), January 26-28, Oita, Japan*, 204-207.

[13] Buller A. & Shimohara K. (2003) Artificial Mind: Theoretical Background and Research Directions, *The 8th International Symposium on Artificial Life and Robotics (AROB 8th '03), January 24-26, Beppu, Japan*, 506-509.

[14] Dautenhahn K. & Nehaniv C.L. (2002) The Agent-Based Perspective on Imitation, In: Dautenhahn K. & Nehaniv C.L. (Eds.) *Imitation in Animals and Artifacts*, A Bradford Book/The MIT Press, 1-40.

[15] Eeckhaut H. & Van Campemhout J. (2003) Handcrafting Pulsed Neural Networks for the CAM-Brain Machine, *The Eight International Symposium on Artificial Life and Robotics (AROB 8th '03), January 24-26, Beppu, Japan*, 494-498..

[16] Freud S. (1920/1990) *Beyond the Pleasure Principle*, W.W.Norton & Co.

[17] Minsky M. (1985) *The Society of Mind*, Simon & Schuster.

[18] Nolfi S. & Floreano D (2000) *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*, A Bradford Book/The MIT Press.

[19] Nowak A. & Vallacher R.A. *Dynamical Social Psychology*, Guilford Press.

[20] Shimohara K. (2002) Human Communication Project: Towards full realization of the richness of natural human communication, *ATR UptoDate, Spring 2002*, 9-11.

[21] Wadsworth B.J. (1996) *Piaget's Theory of Cognitive and Affective Development: Foundations of Constructivism*, 5th edition, Addison-Wesley.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Narrative and Music as Variation: Transformation of Musical Structure Based on Narrative Discourse Theory

Fuminori Kobayashi
Graduate Scholl of Engineering
Yamanashi University
Yamanashi, 400-8511 Japan
g02mk015@ccn.yamanashi.ac.jp

Takashi Ogata
Department of Medicine and Engineering
Graduate School
Yamanashi University
Yamanashi, 400-8511 Japan
ogata@esi.yamanashi.ac.jp

Abstract

This paper proposes an approach of musical structure's transformations (namely, the problem of musical variation) applied the knowledge for narrative structure's transformation developed by Genette's narratology. First, we discuss the correspondences among musical elements and narrative elements, and extract elements in musical structure. Second, we explain concrete techniques of musical variation about some discourse aspects; order, duration, frequency, distance, and perspective. Finally, we show briefly the technical mechanism of an experimental system to implement above techniques. This research is an approach focusing on only structural aspect of narrative and music, and we are thinking of the use as a part of the narrative generation.

1 Introduction

We have studied narrative generation system based on cognitive science and artificial intelligence. Narrative generation process consists of story process, discourse process, and representation process. First two processes relate with conceptual level, but last process relates with surface media level [1]. Expressing media of narrative have various kinds like language, picture, music, body, and so on. We want to grasp it as a medium for expressing narrative itself. However, music is more abstract than language and picture, and is difficult to express semantic information that language and picture have. But, knowledge about the transformation of narrative structure can be commonly used in music and other narrative media. In this paper, we propose methods of musical structure's transformation (namely, the problem of musical variation) applied Genette's narrative discourse theory in narratology. The knowledge for structure's transformation developed by narratology can be applied to the knowledge for musical variation. "Musical narratology" by Eero Tarasti who is a researcher of musical semiotics is similar approach [2]. He adopted the idea of generation trajectory by Greimas [3]. We use Genette's theory [4] as the basic framework of our study because we have adopted it as one of the foundation of our computational modeling of narrative generation. About music, we implemented a prototype of an automatic melody generation system based on simplified theories and knowledge on chord and backing [5, 6], and we use the mechanism in a part of this study too. In relating researches, there is an attempt of the system in which music is generated from molecule structure [7].

Genette's discourse theory [8] is composed of following three aspects; time or tense, mood and voice. Following explanation refers Genette's original description [8] and Prince's summary [9]. Time or tense means the set of temporal relations (speed, order, distance, etc.) between story (narrative content) and discourse or narrated and narrating. Order is one of its subcategories and shows the order in which events occur and the order in which they are recounted. Duration means the relation between story time (narrative world's time) and discourse time (described time). Frequency is the category which means the set of relations between the number of times an event occurs and the number of times it is described. Mood, next major category, is the set of modalities, distance and point of view or perspective regulating narrative information. Distance, first subcategory of two major factors in mood, means whether a narrator shows (the aspect of mimesis) or tells (the aspect of diegesis) narrative information. Namely, showing is taken to institute less distance than telling. Perspective is the perceptual position in terms of which the narrated situations and events are rendered. It is divided into three types; zero perspective, internal perspective, and external perspective. Voice, last major category, treats temporal and spatial relations between a narrator's narrative action and a narrated text.

2 Narrative Discourse Theory and Variation Mechanism

In this section, we describe the basic framework of musical structure's transformation and the concrete techniques. We often call "transformation" "variation", which is a musical term.

2.1 Correspondence among elements in narrative and music

Main elements in narrative are corresponded to musical concepts.

(a) Story, discourse, and narrative discourse theory: An original music is corresponded to a story, and this music is varied or transformed to various structures or forms of a corresponding discourse. We use narrative techniques based on narrative discourse theory in order to process the music transformation.

(b) Narrator: A narrative has generally a narrator as a subject of narration. Each musical instrument (or each player playing an instrument) implemented using a musical programming tool can be a narrator.

(c) Character: We can regard each character as a musical instrument such as narrator. As shown above, each instrument which is regarded as a narrator plays a melody line, but each instrument which is regarded as a character plays a backing line. Different characters have different colors of sound.

(d) Narratee (or receiver or audience): A user corresponds to a narratee. In a literary theory, a narratee is the existence or the concept inside a story world, but we simply regard it as the equal existence with a user outside a story world.

2.2 Elements in musical structure

Here, we classify elements in musical structure for variations into some categories, and the corresponding relations among their elements and narrative discourse techniques. In figure.1, left part shows musical elements and right part shows discourse techniques. First, musical structure is divided into "sequential structure" (lateral structure) and "simultaneous structure" (vertical structure). The former has two categories, "time" and "note". As lower categories, time has "construction", "speed", and "rhythm", and note has "tonality (scale)", which means a sequence of notes. Simultaneous structure is formed by note, and has "harmony" and "tone color" as next level. These musical elements are related to narrative discourse techniques. In sequential structure, construction corresponds to "order" (including "analepsis" and "prolepsos"), "frequency" (including "singularizing", "repeating", and "iteration"). Speed corresponds to "duration" including "pause", "scene", "summary", and "ellipsis". In simultaneous structure, harmony corresponds to "distance". Tone color corresponds to "perspective" and "alteration". The former includes "internal fixed perspective", "internal variable perspective", "internal multiple perpective", and "external perspective". The latter includes "paralipsis" and "paralepsis".

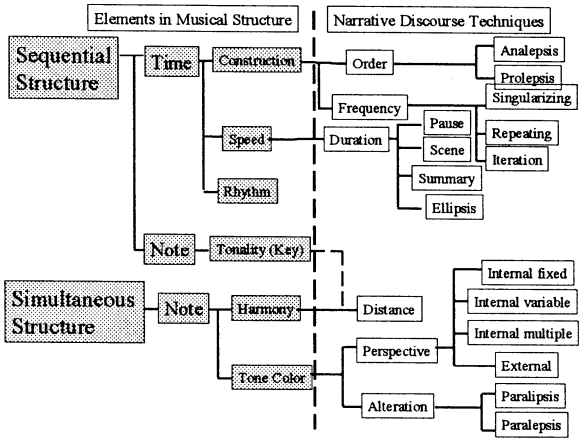


Figure.1: Elements in musical structure and narrative discourse techniques

2.3 Techniques of musical variation

2.3.1 Order

The variation technique of analepsis inserts an already appeared or past musical unit into current point. Prolepsis inserts a not already appeared or future unit into current point. By above two ways, the order a unit appears in an original music (Figure.2).

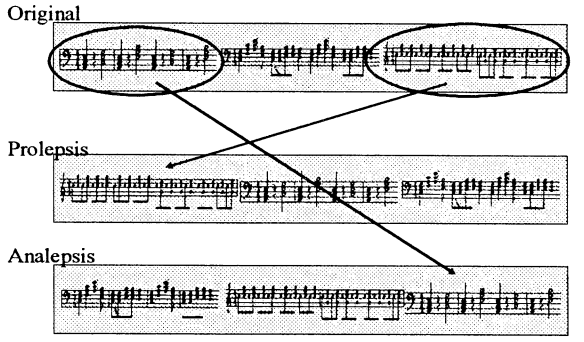


Figure.2: An example of order-transformation

2.3.2 Duration

Duration has four techniques; pause, scene, summary and ellipsis. Pause stops the temporal progress in a story, and adds no description. In scene, the temporal progress in a story is equal to the temporal progress of the discourse. Summary processes discourse's time that has faster speed than story's one. Ellipsis cuts the description of a part in a

story which originally requires a temporal progress.

In the case of music, we call the number showing a musical unit's speed "bpm (beat per minute)". We think that bpm corresponds to the degree of duration, and changing the bpm in each structural unit in a music causes the degree of duration.

At last, in Genette's theory, discourse technique slower than an original story doesn't been defined. As we want to corresponding relation between narrative discourse theory and music, we also do not use the method now. But, the definition of such king of technique will easy in music.

2.3.3 Frequency

This technique binds similar units or repeats one unit in an original music. We regard the frequency in an original music as "singularizing". As shown in figure.3, "repeating" which is a technique in frequency uses repratedly one structural unit in an original music. "Iteration" binds two structural units into one using two concrete techniques. First one plays simultaneously two units. Second one synthesizes two pitches of notes by taking the average of two notes-numbers, which is established by MIDI standard.

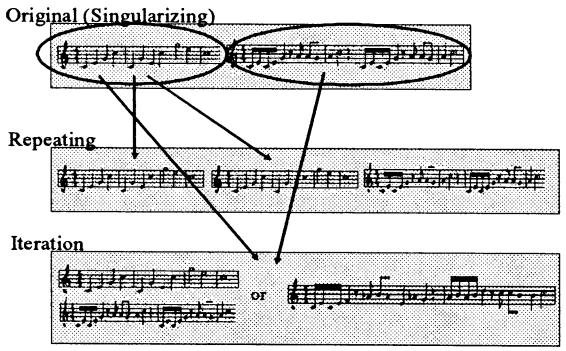


Figure.3: An example of frequency-transformation

2.3.4 Distance

In this study, a narrator corresponds to backing playing with a musical instrument defined beforehand, and the processing of distance is defined by operations of a narrator's backing. The processing of other instruments is related to next perspective's processing. Distance is controlled by two kings of musical operations; narrator's backing and the degree of narrator's intervention into a narrative. The farmer's concrete processing is currently modulation, and nearer distance and farther one makes more strong modulation to an original backing. In the latter, nearer distance decreases the degree of narrator's intervention and farther distance increases one. In the most near distance, no intervention is done, and narrator's backing disappears. On the contrary, in the farer distance, by extremely high intervention, only narrator's backing is played without any other playings.

2.3.5 Perspective

As performing instruments correspond to characters in this research, perspective is operated by changing the kind of musical instruments to be distinguished in note color. The techniques of perspective select the range to be narrated. The range includes internal aspects (mental states and so on) and external aspects (actions, external characteristics, and so on) in characters. In this research, an external aspect is corresponded to the backing part performed by an instrument, and an internal aspect is corresponded to the melody part performed by an instrument. In the case of internal fixed perspective, next three elements are played; the internal aspect, namely the melody in a focused instrument, and external aspect, namely the backing in a focused instrument, and the external aspect in other instruments which appear in the structural unit treated currently. On the other hand, in the case of external perspective, external aspects of musical instruments which appears in the unit treated currently, namely backing is (or backings are) performed. Next, internal variable perspective changes focused musical instruments. Last, internal multiple perspective simultaneously focuses on two or more instruments.

There is another category called alteration which means a temporary transgression to a mainly adopted type of perspective. Alteration has two sub-categories; paralipsis and paralepsis. In the paralipsis, narrative information to be reported theoretically is not intentionally narrated and in the paralepsis, narrative information to be ignored theoretically is intentionally narrated. We can introduce these techniques into music by the application of above musical techniques of perspective.

3 An Experimental Variation System

We are implementing a musical variation system on a Macintosh personal computer with Max/MSP, to attempt these variation techniques by real sound. The system has MIDI files of the original music which is the source for variation. The MIDI files are divides into 31 kinds of backing files and 7 kinds of melody files. The length of each MIDI file is 4 measures. Figure.4 shows the outline of the system. Variation techniques explained in this paper are used in the part of "variation system" in this figure. The processing of "original music composer" is related to the whole framework of this research on which we don't refer in this paper. Namely, this research's extreme plan is developing a system which generates a narrative text from music and music from a narrative text based on structural and other transformation techniques between narrative and music. Therefore, "original

music" in figure.4 is one which is composed from a narrative text. Currently, we have developed an original music based on Propp's narratology [10], and fragments of backing lines and melody lines prepared preliminarily are related with structural aspects of the Propp's narrative text. Each character in Propp's narrative has a melody line such as a theme and each function has a backing line, and their combinations compose an original music. Detailed explanation of this mechanism will be shown in other papers.

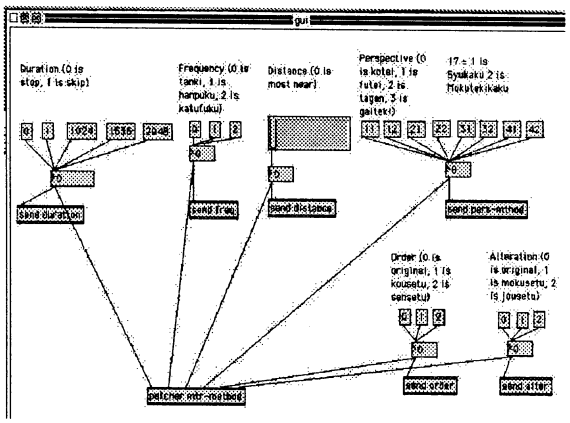


Figure.4: A screen of GUI in Max/MSP

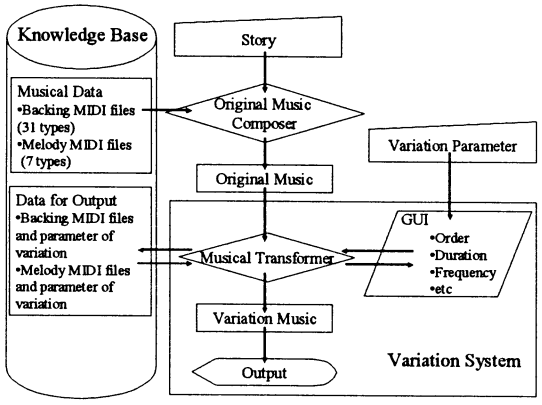


Figure.5: Outline of the system

4 Conclusions

In above description, after we showed correspondences between narrative's concepts and music ones and classified musical elements in musical structure's transformation, we proposed mechanisms of musical variation techniques defined using Genette's narrative discourse theory and its precise and computational re-definition by our computational narratology. At last, we showed the outline of the musical variation system.

We treated the problem of structural transformation in narrative and music in a same framework. Of course, music is more abstract than language and picture, and it is difficult to express by music such information that language and picture have. In the sense, this attempt may be an aspect of analogy or a kind of contortion. But, if we can define a consistently structural relation of transformation between narrative and music, we can make both mechanisms of music generation from narrative and narrative generation from music or a mutual cyclic transformation mechanism between narrative and music in which music is not only an accompaniment of a narrative.

References

[1] Ogata, T., "Expanded literary theory: Cognitive /computational expansion of literary theories and narratology," *Proceedings of the 17th Congress of the International Association of Empirical Aesthetics*, 4 (8), pp.163-166, 2002.

[2] Tarasti, E., A Theory of Musical Semiotics, Indiana University Press, 1994.

[3] Greimas, A. J., Semantique Structurale: Recherche de Methode, Larousse:Paris, 1966.

[4] Ogata, T. & Yazawa, K., "On the Rhetoric for Narrating the World," Yoshimine, N., et.al. (eds), *Literature in Cognition and Computer2: The Expansion of Literature, The Japanese Cognitive Science Society Technical Report*, 40, pp.10-13, 2000.

[5] Kobayashi, F. & Ogata, T., "Automatic Melody Composition Based on the Constraint of Chord and Music Knowledge," *Proceedings of the 17th Congress of the International Association of Empirical Aesthetics*, 4 (8), pp.565-568, 2002.

[6] Kobayashi, F. & Ogata, T., "Musical Generation by Transformation –Toward the Narratology of Music," *Abstracts of the 4th International Conference on Cognitive Science*, pp.92, 2003.

[7] Fujimoto, T. & Takahashi, Y., "Molecular Music: Listen to the Sounds from Molecules, (In Japanese)" *The 17th Annual Conference of the Japanese Society for Artificial Intelligence*, 2003.

[8] Genette, G., Discours du recit, essai de methode, In Figures III, Seuil:Paris, 1972.

[9] Prince, G., A Dictionary of Narratology, University of Nebraska Press, 1987.

[10] Propp, V., Morphology of the Folktale, University of Texas Press, 1968.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Autonomous Mobile Cars on a Highway Controlled with Neural Networks

Tsuyoshi Uemura*, Takahiro Kai*, Tatsuro Shinchi**, Masahiro Yokomichi* and Tetsuro Kitazoe*
* Faculty of Engineering, **Faculty of Education and Culture, University of Miyazaki
Gakuen Kibanadai Nishi 1-1, Miyazaki City, Japan, 889-2192
kai@cs.miyazaki-u.ac.jp

Abstract

This paper presents a simulation model of multiple autonomous mobile robots, in which neural networks are employed. We used numbers of inch robot Khepera to simulate mobile cars running on a highway. Autonomous behavior is assumed for the robots to run freely by sensing other neighboring robots or guardrails of the road by means of infrared ray. Genetic Algorithms(GA) were applied to develop a driving technique by coordinating left or right wheel speed to sensor signals. Neural networks were used to control left or right wheels connecting to input sensor signals with appropriate weight. GA was applied to determine 8 real coupling weights.

As a result of optimization of behavior models, a few of nice running models were generated for 10 robots going around a circuit way. A variety of cases were tested where a road was curved, dented, or blocked, for instance, by a car with engine trouble. We obtained a nice collective movement of multi-robots, with high techniques, such as speeding up, slowing down, and overtaking a slow robot ahead.

1 Introduction

Many kinds of experimentations for driverless robots have been done so far to attain safe and effective transportations. It will be our dream if we have truly autonomous robots which have a free style of driving; every autonomous robot on a road has its own way of driving and can choose its own speed. In the researches carried out so far about Intelligent Transportation Systems (I.T.S.), some systems have succeeded in realizing collective movement of vehicles. Those movements, however, are quite rigid; vehicles run by simply keeping a good distance from the robot ahead, or by only following markers embeded in the road. Namely, individuals run just in file.

We employ 10 Khepera robots running along a road with two lanes which makes a circuit with some curves or narrow spaces. A robot emits infrared rays and receives reflected signals from other neighboring robots or guardrails which stand at the sides of a road. We use neural network models to control left or right wheels by receiving sensor signals.

A simulation of these multi-robots system causes a serious problem for a computer with one *CPU*. Recent unix workstation (Ultra 60,Solaris 8) enabled us to simulate more than 10 Khepera robots at the same time. It is, however, not easy to develop genetic algorithms(GA) for these 10 Khepera robots. It will take almost one day if one wants to decide 10 real parameters by GA. In the present paper we use 8 real parameters which are synaptic couplings attached to each sensor signal. However,it is hopeless to decide these real parameters if the ranges of their values are indefinite. Fortunately we can estimate ranges of their values by refering the results of the previous work which was obtained from a fish school model of khepera.[1] By applying GA for synaptic couprings, eventually we obtained a beautiful movement of multiple robots along a road without any accident, even if a road has narrow or curved spaces.

2 Experimental Environment

2.1 Mobile Robot and Simulator

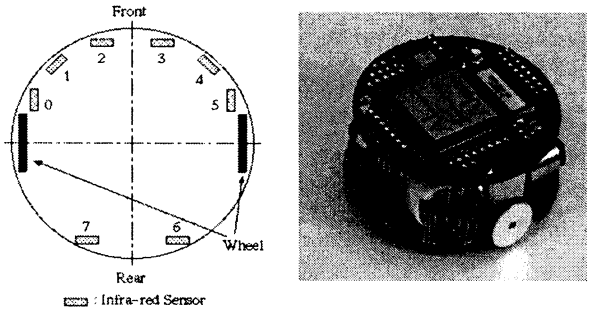


Fig. 1: *Mobile Robot Khepera*

In our experiments, we assume to use a miniature mobile robot Khepera. [2] Sensors of Khepera detect an obstacle and return values in integer between 0 and

1023 corresponding to the distance between the Khepera and the obstacle. Low value means that there are no obstacles near the sensor, while high value means that an obstacle is close to the sensor. The Khepera communicates with a computer through a serial line, so that the computer obtains the sensor values from the Khepera and provides the wheel speed to the Khepera.

We investigate robot movement in computer simulative model to verify rules for behavior. *Khepera Simulator ver.2.0* written by Olivier Michel (downloadable from the *WWW* [3])is an excellent software by taking account of uncertainty which is supposed to exist in the real world.

2.2 Conditions in the Simulator

We set up the various kinds of roads in our simulations. An example is shown in Fig.2, where one is the straight road and another is the curved road. Two roads in Fig. 2 are linked each other something like a circuit. Therefore, robots run repeatedly on the looped way of a) and b) in Fig.2.

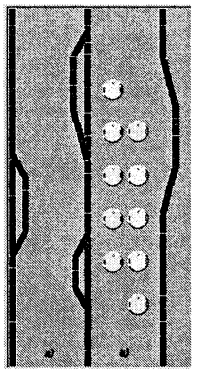


Fig. 2: *The road in the simulations.The width of the road is 3 times of the diameter of Khepera.*

They are divided into two types of robots with the different speed, where fast 5 robots have maximum speed V_{max}, while slow 5 robots have maximum speed $3V_{max}/4$. Each robot frequently changes its direction according to the returned values from the installed 8 sensors, detecting obstacles of neighboring robots or guardrails set on both sides of a road. The behavior models of a robot should be optimized for the smooth movements of multi-robots.

3 Behavior Models of A Robot

In the previous work we applied the behavior model of a fish. The behavior model of a fish school was modeled by I Aoki [4] and proved to have a nice coincidence with experimental results. Sensors equipped on Khepera should be used like as the sense organs of a fish, such as eyes or lateral lines. Therefore, rear sensor ♯ 6 and ♯ 7 are not used for interactive rules between robots. While the distance r determines a behavior type in fish school models, s_x returned from sensor ♯x determines the behavior of Khepera. To simplify our models, two sensors of s_0,s_1 and s_4,s_5 are coupled as $0.5 \cdot (s_0 + s_1)$ and $0.5 \cdot (s_4 + s_5)$, respectively. The region which determines the behavior type of a robot is divided concentrically with boundary sensor values (S_1,S_2, S_3) as shown in Fig.3. Tab.1 is the behavior type of a Khepera robot translated from a fish school model[4]. Since a robot is required to avoid collisions, the largest value among returned sensor values ($s_{0,1}, s_2, s_3, s_{4,5}$) is chosen to determine the robot behavior.

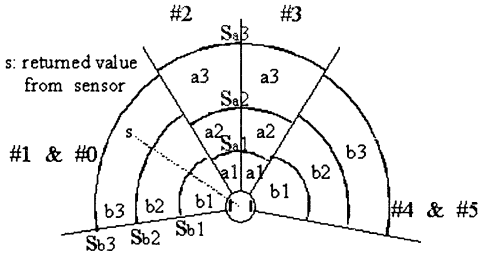


Fig. 3: *Discernible Rages of Khepera Sensors. The region which determines the behavior type of a robot is divided concentrically with sensor values (S_1,S_2,S_3). Sensor of* ♯2 *or* ♯3 *detects obstacles ahead area denoted as a with radius S_{a1}, S_{a2}, S_{a3}. Sensors of* ♯0&♯1 *and* ♯4&♯5 *detect lateral area denoted as b with S_{b1}, S_{b2}, S_{b3}, respectively. The distance to an obstacle is measured with the returned senor value s.*

area	*a)* ♯2, ♯3	*b)* ♯0&♯1, ♯4&♯5
$s \geq S_1$	quick turn	quick turn
$S_1 > s \geq S_2$	decelerate	accelerate
$S_2 > s \geq S_3$	accelerate	slight turn
$S_3 > s$	accelerate	accelerate

Tab.1: *The robot behaviors transferred from a fish school model*

GA was used for 700 generations with each 2500 steps run for 25 robots. 10 offspring are selected to make 15 new robots with 25% mutation rate. The result is shown in Tab.3.

The evaluation function in our experiment is given as

$$g = \frac{Distance_y}{Collision_{robot} + Collision_{guardrail} + 1} \quad (1)$$

where $Distance_y$ measures how long a robot moves forward (upward in Fig.2) in a trail. $Collision_{robot}$ and $Collision_{guardrail}$ mean the number of collision against other robots and guardrails, respectively.

After the genetic algorithm was applied, we obtained 5 typical parameters which were shown in Tab.2.

	S_{a1}	S_{a2}	S_{a3}	S_{b1}	S_{b2}	S_{b3}	g
1	600	271	20	624	110	19	25.515
2	537	425	16	668	163	93	24.755
3	537	491	18	707	173	123	24.653
4	941	720	599	668	492	312	23.847
5	941	720	156	697	624	528	23.744

Tab.2: $S_{a1} \sim S_{b3}$ *of the best 5 robots after 100 generations.*

We proceed to decide driving types by GA after the discernible regions were determined. We use the best one in Tab.1 which is fixed here after. We use discretized wheel speeds for GA application by dividing maximum speed V_{max} by 4. A driving type is, for example, denoted as (3,2), meaning that left and right wheels have speeds with $3V_{max}/4$, $2V_{max}/4$, respectively. Driving type (i, j) are used as genes and GA is applied for the genes in each area. Since sensors on Khepera are equipped symmetrically, the driving types in the area for ♯3, ♯4 & ♯5 are taken just opposite to ♯2, ♯0 & ♯1, respectively. If the highest sensor value is lower than S_{b3}, S_{a3}, then slow and fast robots take $(3,3)$ and $(4,4)$, respectively. The 10 robots are placed at each trial in the right side road as shown in Fig.2. GA was applied with the same conditions as before.

best robot		a_1	a_2	a_3	b_1	b_2	b_3	#of no accident
①	fast	0,0	2,3	2,4	3,0	4,3	4,3	28
	slow	1,0	1,3	1,3	3,0	3,3	3,3	
②	fast	0,0	2,3	0,2	3,0	4,4	4,3	25
	slow	1,1	1,3	1,3	3,0	3,3	3,3	
③	fast	0,0	1,3	0,2	3,0	4,4	4,3	23
	slow	0,0	2,3	1,3	3,0	3,3	3,3	
④	fast	0,0	2,4	1,2	3,0	4,4	4,3	20
	slow	0,0	0,3	0,3	3,0	3,3	3,3	

Tab.3: *Driving types*(V_{left}, V_{right}) *in* ♯0&♯1 *and* ♯2 *areas determined by GA.* ♯ *of no accident means the period while a group of robots survived continuously without any accidents as the best number, through generation proceedings.*

4 Controls by Neural Networks

Next, neural networks were used to determine wheel speed by connecting input sensor signals to wheel speed controllers with appropriate synaptic coupling weights. A Khepera runs straight with full speed if there is no obstacles nearby with receiving no sensor signals. Therefore it seems natural to consider that the neural networks from sensors are connected to the neuron to control the braking of each wheel. Thus, the wheel speeds of left and right wheels L, R are expressed by using the simplest neural network as

$$L^f = V_f(1 - f(\omega_1 x_1 + \omega_2 x_2 + \omega_3 x_3 + \omega_4 x_4 + \omega_0)), \quad (2)$$
$$R^f = V_f(1 - f(v_1 x_1 + v_2 x_2 + v_3 x_3 + v_4 x_4 + v_0)) \quad (3)$$

for fast robots and

$$L^s = V_s(1 - f(\omega_5 x_1 + \omega_6 x_2 + \omega_7 x_3 + \omega_8 x_4 + \omega_9)), \quad (4)$$
$$R^s = V_s(1 - f(v_5 x_1 + v_6 x_2 + v_7 x_3 + v_8 x_4 + v_9)) \quad (5)$$

for slow robots, where second terms in Eqs.(2)$\sim$(5) represent the controls of braking speeds. Here x_1, x_2, x_3, x_4 are sensor values for ♯0&♯1, ♯2, ♯3, ♯4&♯5 normalized as 0$\sim$ 1.0 defined by $x_i = s_i/1023$ and ω_i, v_i are synaptic couplings. $f(x)$ is a sigmoid function defined as $f(x) = (1 + \tanh(x))/2.0$. If there is no priority for right or left lane moving in our simulation, it is natural to assume that a robot has left–right symmetric behavior. If left, right wheel speeds and left, right sensors are exchanged in the neural network equations at the same time, we will have the same situation. This means that $\omega_1 = v_4, \omega_2 = v_3, \omega_3 = v_2, \omega_4 = v_1, \omega_5 = v_8, \omega_6 = v_7, \omega_7 = v_6, \omega_8 = v_5$. We also set the threshold values as $\omega_0 = v_0 = \omega_9 = v_9 = -1$ to fix the maximum speeds for robots when there are no sensor values returned. Thus, we have 8 chromosomes $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6, \omega_7, \omega_8$ to be determined if we want to apply GA to Khepera robots for efficient movement on a high way.

If the ranges of the real parameters $\omega_1 \sim \omega_8$ are indefinite, we will face with a serious difficulty in applying GA. Fortunately, however, we can estimate the ranges from the results of the previous fish school model. For instance, Tab.3 shows that the right wheel speed for fast robot changes wheel speed as 4,3,3,0

when x_1 moves from 0 to 1,where other $x_i (i \neq 1)$ are set to 0. The result is compared with the neural network in eqs.(3) by setting $x_2 = x_3 = x_4 = 0$.

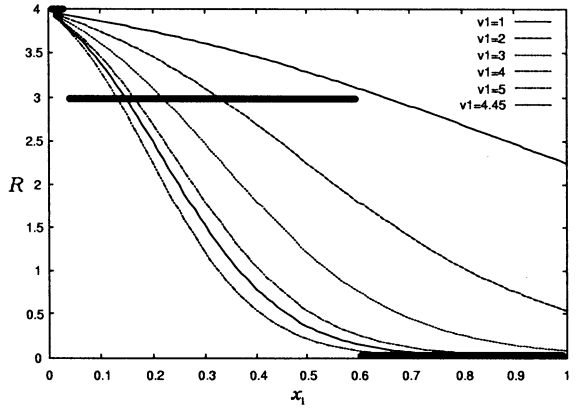


Fig. 4: *Right wheel speed for ♯0&♯1 sensor in Tab.3 is compared with neural network(3) by changing v_1 where V_f=4.5. v_1=4.45 is the result of GA shown in Tab.4*

Fig.4 shows that the ranges of $v_1(\omega_4)$ will be $v_1 = 1 \sim 5$. In the similar way we can estimate ranges of ω_1, ω_5=0 $\sim$ 1, other ω_i=0 $\sim$ 5($i \neq 1, 5$).

GA was applied similarly as before. After more than 100 generations, we obtained two kinds of ω eventually as shown in Tab.4.

	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8
①	0.24	5.00	0.25	4.45	0.14	4.30	0.55	2.50
②	0.24	5.00	0.25	4.45	0.14	4.30	0.90	2.50

Tab.4: *The results of synaptic couplings where V_f=5.5,V_s=3.5.*

The evaluation function for best 10 robots proceeded toward 14 as shown in Fig.5 and ω_i were obtained as $v_1(\omega_4)$ in Fig.6. A Test run was performed for 50 trials by changing the initial starting configuration for each trial. We obtained a smooth movement without any stagnation for ① robots and 1 trafic stagnation for ② robots.

5 Conclusion

The neural network was applied to control the braking of each wheel according to sensor values. Amazingly, this scheme gives very stable movement for multiple robot system. If ω_i are changed in a range, the robots are stable with high evaluation value as seen from Fig.5,6, which stands a contrast with rather fragile fish school model given in section 3.

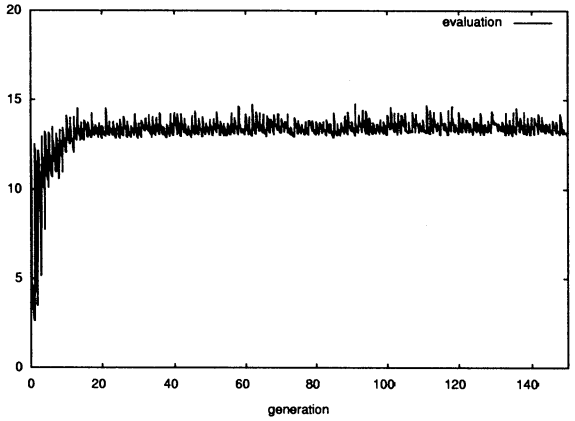


Fig. 5: *Evaluation function in GA procedure for best 10 robots with V_f=5.0,V_s=3.5.*

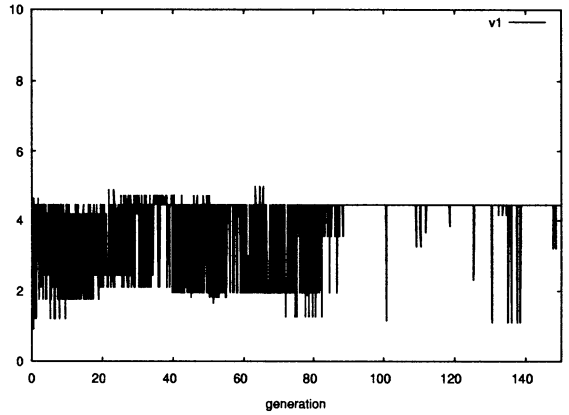


Fig. 6: $v_1(\omega_4)$ *in GA procedure for best 10 robots with V_f=5.0,V_s=3.5.*

References

[1] T.Shinchi, M.Tabuse, A.Todaka, T.Kitazoe; Collective movements of mobile robots with behavior models of a fish, Proc. of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, vol.1, p.p. 365-369(2002)

[2] K-Team SA: Khepera USER MANUAL Version 4.06. (1995).

[3] Michel, O.: *Khepera Simulator* Package version 2.0. Freeware mobile robot simulator. Downloadable from the World Wide Web at `http://diwww.epfl.ch/lami/team/michel /khep-sim/`

[4] 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)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Stereo Matching from Color Images Based on Self-organization Neural Networks

Xijun Hua
Faculty of Engineering
Miyazaki University
Miyazaki, 889-2192, Japan

Masahiro Yokomichi
Faculty of Engineering
Miyazaki University
Miyazaki, 889-2192, Japan

Michio Kono
Faculty of Engineering
Miyazaki University
Miyazaki, 889-2192, Japan

Abstract

In this paper, we use two layered self-organization neural network model to simulate the competitive and cooperative interaction of binocular neurons. We define a special similarity function as initial input to make full use of the color information. The similarity map is established by taking logical AND calculation of red, green and blue color value similarity. We first consider color random-dot stereogram for stereo correspondence. For the real images we propose a segmentation method to deal with the similarity map. Experiments results have showed that the quality of the matching results can be considerably improved comparing with the conventional grey value algorithm.

Keywords: stereo vision, stereo matching, disparity, RGB color space, neural network, self-organization

1. Introduction

Stereo vision is a well-known technique for obtaining depth information from stereo digital images. As one of the main components of computer vision, stereo vision is a process to reconstruct the 3D shape of visible surfaces by measuring disparity of two corresponding points in a stereo image pair. It is a direct and passive method of obtaining depth information of the visual world which makes it attractive for applications like 3D reconstruction, object recognition, navigation and manipulation.

The Stereo matching, or stereo correspondence problem, which is to identify corresponding points in two or more images of the same scene, has become the most important and difficult issue of stereo vision and it is an ill-posed problem. So far, various correspondence algorithms such as feature-based, area-based, phase-based, energy-based, multiple-baseline and data fusion have been studied.

There are some limits for existing approaches to recover precise and dense disparity map. For a successful reconstruction of complex scene it is essential to obtain dense disparity maps defined for every pixel in the entire image. Therefore, the neural network techniques to conventional stereo vision problems have received increasing attention in recent years. The importance of competitive and cooperative neural network was shown through interactions mediated by excitatory and inhibitory axon collateral pathways in rat visual cortical slice[1]. Amari and Arbib (1977) proposed a famous neural network model for the stereo matching[2]. A neural network model for solution of the correspondence problem was proposed by Marr, which is composed of groups of neurons selectively responsive to different disparities of visual stimuli shown to the two eyes[3]. Reimann and Haken (1994) proposed a new mathematical model to solve the correspondence problem by a dynamical self-organizing process[4]. Kitazoe et al. (1998) researched sequential stereo vision with hysteresis in competition and cooperation neural network[5].

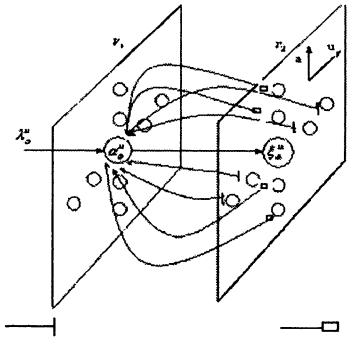
In our past work, we only used the intensity information of the stereo images for stereo matching[6]. Usually, in stereo system, the digital cameras capture color images, which are afterwards translated into intensity (grey value) images for further processing. In the literature, most of the stereo matching algorithms have been limited to grey level images, only few authors have dealt with color images. Actually, a color pixel is multi-valued (3 bytes, one for each band), while in the associated intensity image the information content is single-valued (1 byte). The color information is usually neglected, but undoubtedly it may be of great use in helping a matching system perform better. It has been reported that the precision of matching results can be improved by 20 to 25% when using the adaptive color algorithm instead of the grey value algorithm [7].

In this paper, in order to improve the accuracy of disparity map we proposed a new color stereo matching approach to deal with color images straightly. The paper is organized as follow: In section 2, we introduce two layered self-organization neural network model and define a special similarity measure function to establish R, G, B similarity map. In section 3 and 4, Effect of the proposed model is examined by experiments with both random color stereogram and real images. A segmentation method is proposed to deal with the input similarity maps of real images.

2. Neural networks by self-organization

The neural network model is shown as Fig.1. a represents range of disparity, and u represents pixel position. The binocular neurons compete over the disparity search area and cooperate over the cooperation area in such a

manner that in the course of time just one binocular neuron win; all others end with zero, therefore, resulting in a stereo disparity perception.



Excitatory connection Inhibitory connection

Fig. 1 Neural network with two layers (1D model)

The neural network equations are written as eqs. (1) - (6). α_{ab}^{uv} is membrane potential of neuron. ξ_{ab}^{uv} is output membrane potential of the neuron. *A*, *B* and *C* are positive constants which have to be chosen appropriately.

$$\tau_1 \dot{\alpha}_{ab}^{uv} = -\alpha_{ab}^{uv} + A\lambda_{ab}^{uv} - B\sum_{a' \neq a} g(\xi_{a'b}^{uv}(t)) + C \sum_{u' \in U_u, v' \in V_v} g(\xi_{ab}^{u'v'}(t)) \quad (1)$$

where U_u and V_v denote certain cooperation area of u and v respectively.

$$\tau_2 \dot{\xi}_{ab}^{uv}(t) = -\xi_{ab}^{uv}(t) + f(\alpha_{ab}^{uv}) \quad (2)$$

$$\lambda_{ab}^{uv}(R) = 1 - |R_l(u,v) - R_r(u+a, v+b)| \quad (3)$$

$$\lambda_{ab}^{uv}(G) = 1 - |G_l(u,v) - G_r(u+a, v+b)| \quad (4)$$

$$\lambda_{ab}^{uv}(B) = 1 - |B_l(u,v) - B_r(u+a, v+b)| \quad (5)$$

here $\tau_1 << \tau_2$, $g(x)$ is a function given as:

$$g(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (6)$$

and $f(x)$ is a sigmoid function. In eq. (1), *a* and *b* stands for the epipolar and the perpendicular disparity. The input data λ_{ab}^{uv} are obtained as the product of R(red), G(green) and B(blue) similarity function (3), (4), (5). $R(u, v)$, $G(u, v)$ and $B(u, v)$ present the color intensity value of pixels respectively. In eq.(1), α_{ab}^{uv} has inhibitory connection for a competition(the third term) and excitatory connection for a cooperation(the fourth term). For simplicity, we consider the case where the two eyes are located horizontally so that we can neglect "b" in eqs. (1)-(5).

3. Stereo matching for random color stereogram

Each band of the color stereogram is generated in a manner similar to the conventional random dot stereogram. Therefore, the colors of the dots are randomly chosen. The random stereogram is composed of 20 scan-lines. Every scan line is composed of 80 random dots, in which from 25th to 64th dots, the relative disparity to the background is given as 5. The R, G, B similarity map are shown in Fig.2-4, respectively. The AND similarity map is shown as Fig.5. In Fig.5, the potential corresponding points with high similarity value become fewer than the R,G,B similarity map.

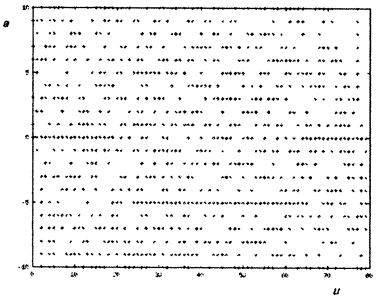


Fig. 2 Similarity map (Red)

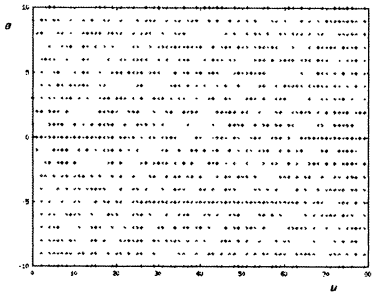


Fig. 3 Similarity map (Green)

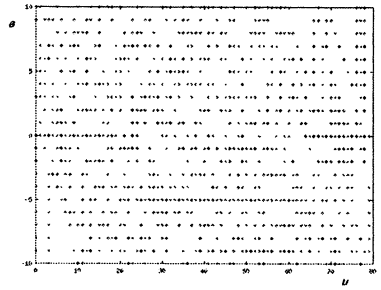


Fig. 4 Similarity map (Blue)

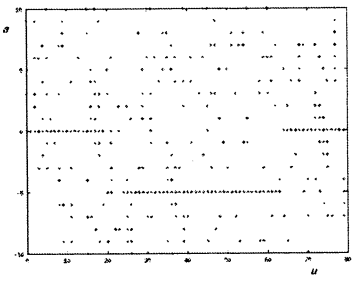


Fig. 5 Similarity map (AND)

From Fig.6, a satisfactory disparity map is obtained. Here, Loop represents essential circulating times of program for convergence. Though the images are synthetic and free of noise, there are a few mismatches at the left corner due to occlusion.

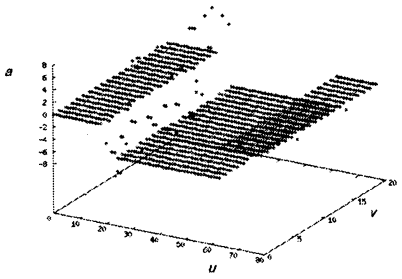


Fig. 6 Disparity map (Loop=70)

When we use the conventional grey value matching algorithm to deal with the grey value images, its similarity map is shown as Fig.7. As shown in its disparity map (Fig.8), the mismatching points are caused by both occlusions and false targets in the boundary region. In addition, the convergence speed of the color matching is faster than the grey value matching.

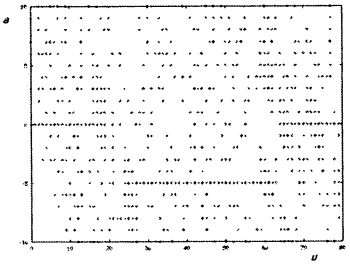


Fig. 7 Similarity map

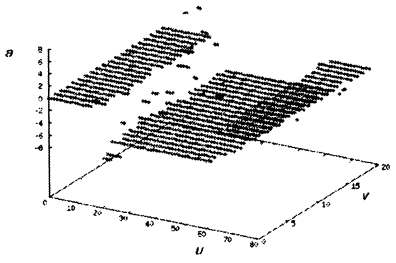


Fig. 8 Disparity map (Loop=95)

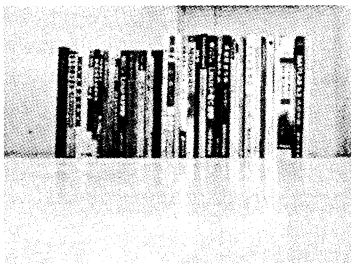
4. Experiment for real images

Here we only consider the parallel cameras case, in which corresponding points in the two images lie on the same horizontal epipolar lines. Therefore, it is necessary to keep two camera axes parallel and the baseline of stereo camera system is at a right angle to the camera axes. In this case, the situation $a > 0$ will not occur.

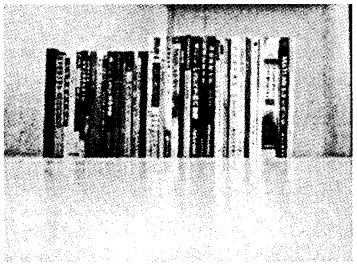
In RGB color space, the R, G, B color images are taken by CCD camera with 640×480 pixels. The element values of similarity map vary in the range [0, 1] because the intensity values of different pixels is always varying in the range [0,255]. In the other hand, there are many factors such as noise which will make precise correspondence computation more difficult than the matching of random stereogram. Therefore we propose a segmentation method to deal with the similarity map. A threshold function (7) for filtering the similarity map is used so as to get binary similarity map. The parameter S is very important to get a satisfactory similarity map. If S is too large or too small, there may be miss-correspondence problems appeared. S has to be chosen appropriately according to the actual real images.

$$\lambda = \begin{cases} 1, & \lambda_i > S \\ 0, & \lambda_i \le S \end{cases} \qquad S : \text{Threshold gate} \qquad (7)$$

The real image is shown as Fig. 9. In the visual scene, two layer books are placed on the table. We expect percept the disparity values of different layers of the books.



left image



right image

Fig. 9 Real images

For simplicity we select the middle area (120×20) of real image to discuss. The R, G, B AND similarity map is shown in Fig.10. From Fig.11, by using AND similarity map, we can obtain satisfactory disparity map in which the ambiguity between potential matches can be obviously reduced.

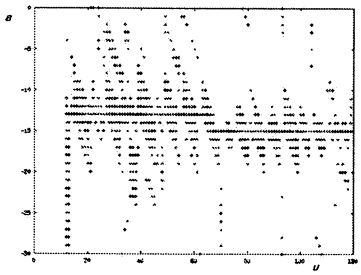


Fig. 10 Similarity map (AND)

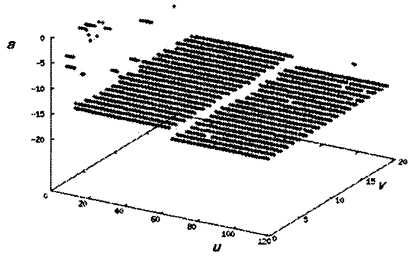


Fig. 11 Disparity map (S= 0.9, Loop=85)

When we used the single grey value matching method, the similarity map is shown in Fig.12. Its disparity map is given in Fig.13, in which there are more mismatches appeared than Fig.11.

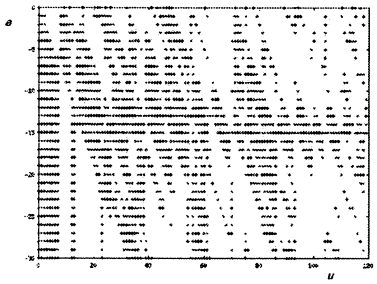


Fig. 12 Similarity map

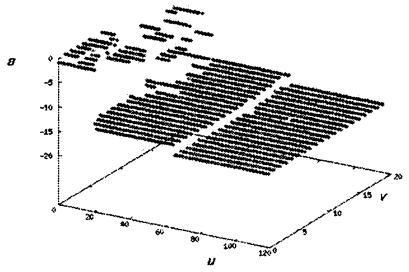


Fig. 13 Disparity map (S= 0.9, Loop=150)

5. Conclusion

In this paper, we use two layered self-organization neural network model to simulate the competitive and cooperative interaction of binocular neurons. We define a special similarity function by taking logical AND calculation of red, green and blue color value similarity. Experiments with both synthetic random-dot stereogram and real images have showed that the disparity quality as well as the convergence speed can be considerably improved by using color stereo matching technique comparing with the conventional grey value algorithm. For the real images, it is necessary to use the segmentation method to deal with the similarity maps. In the future work we will continue to improve our present method and investigate more reliable stereo matching approaches based on appropriate color matching features of stereo images.

References

[1] K. Toyama, M. Tanifuji, "Imaging a computational process in the visual cortex", *Neural Networks,* Vol. 9(8), pp. 1351-1356, 1996.

[2] S. Amari, M. A. Arbib, "Competition and cooperation in neural nets", Systems Neuroscience. Academic Press, New York, pp. 119-165, 1977.

[3] D. Marr, T. Poggio, "A computational theory of human stereo vision", Proc. R. Soc. Lond. B. 204, pp. 301-328, 1979.

[4] D. Reiman, H. Haken, "Stereo vision by self-organization", Biological Cybernetics, Springer-Verlag, pp. 17-26, 1994.

[5] T. Kitazoe, J. Tomiyama, Y. Yoshitomi, "Sequential stereoscopic vision and hysteresis", *Proceeding of 5th International Conference on Neural Information Processing,* pp. 391-396, 1998.

[6] X. Hua, Y. Tang, M. Yokomichi et al., "3-D perception for monochromatic surface by self-organization neural network", *Proc. of the 8th International Symposium on Artificial Life and Robotics*, Beppu, Oita, Japan, pp. 479-483, 2003.

[7] A. Koschan, V. Rodehorst, K. Spiller, "Color stereo vision using hierarchical block matching and active color illumination", *Proc. of 13th international conference on pattern recognition ICPR'96*. Vienna, Austria, Vol. I, pp. 835-839, 1996.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Stereo correspondence by SVM regression

Hiroki Sugiyama
Faculty of Engineering
Univ. of Miyazaki
Miyazaki, Japan

Masahiro Yokomichi
Faculty of Engineering
Univ. of Miyazaki
Miyazaki, Japan

Xijun Hua
Faculty of Engineering
Univ. of Miyazaki
Miyazaki, Japan

Michio Kono
Faculty of Engineering
Univ. of Miyazaki
Miyazaki, Japan

Corresponding E-mail : yokomich@cs.miyazaki-u.ac.jp
Key Words : Stereo correspondence, SVM regression, similarity function

Abstract

Reconstructing 3D scene from multiple images is one of the important tasks for mobile robots. In this paper, the authors propose a SVM (Support Vector Machine) -based stereo matching method. In order to reduce the number of the miss-leading (pixel-disparity) pairs, the similarity function is extended. Its effectiveness is shown with the experiment about real image example.

1 Introduction

Reconstructing 3D scene from multiple (especially two) images is one of the central themes in the computer and robot vision. If we can identify two points in each image and if several camera parameters are known, the 3D position of the point can be calculated by geometrical relations.

Marr and Poggio proposed the calculation theory of stereo vision and various techniques have been studied[1],[2]. The one of the most difficult problems in the stereo vision is the *stereo matching* itself. There are many reasons: Due to the occlusion or illuminative condition, no corresponding points may exist. On the other hand, by means of the texture of the surface, many candidates of the corresponding point may exist. Thus, this problem is essentially one of the *ill-posed problems*.

Amari and Arbib proposed a neural network model for the stereo matching. Poggio considered the stereo matching problem as a kind of optimaization problem and solved it by means of the regularization theory. In [6], the original images are transformed to the feature image by second order derivative filter, and zero-crossing points are used as the candidate of the corresponding points in order to reduce the size of the problem. Then, defining *similarity function* as the difference of the intensities between the candidates of the corresponding pixels, stereo matching is solved as an optimaization problem for the weighted sum of the similarity function and the norm of the disparity function.

However, for several situations, no necessary zero-crossing points can be obtained, and the performance functional is nonlinear, thus, the transformed problem itself is not easy to solve [2].

In the previous work[10], the authors proposed a procedure which transforms the stereo matching problem to a kind of SVM (Support Vector Machine) regression one. In [10], the (pixel-disparity) pairs that has low similarity are deleted based on threshold value. In addition, the number of the isolated pairs that causes the miss-leading and consume the computational costs are decreased by means of a filter which is based on certain continuity assumption on the 3D objects. The method was examined with one and two dimensional random dot stereogram examples.

In this paper, the approach in [10] is examined with real images with many colors. The stereo image pair for real images have many ambiguities because of illuminational conditions, the complexity of the objects, and so on. The authors modify the approach in [10] in the following way. Firstly, the similarity function is extended to the block based one. This has the similar effect as the filter in [10]. Secondly, the selection of the examples with high similarity is applied. Then, SVM regression problem is solved. The effectiveness is examined with real image example.

2 Stereo matching as SVM regression

2.1 Brief Preview for Support Vector Machine Regression

The support vector machine learning was initially developed by Vapnik for the classification problem by linear hyperplane. It was extended to nonlinear classifier by kernel method and to regression problem such that it can learn continuously valued output data.

In the nonlinear regression problem, a training set consists of N examples (x_i, y_i), $x_i \in \Re^n$, $y_i \in \Re$. These input-output data are approximated by nonlinear function with kernel K,

$$f(x;b,c) = \sum_{i=1}^{N} c_i K(x_i, x) + b,$$

and the parameters (b,c) are determined such that the functional

$$H(b,c) = \sum_{i=1}^{N} |y_i - f(x_i;b,c)|_\varepsilon + \lambda \|f\|^2 \qquad (1)$$

is minimized, where $\lambda > 0$ and $|\cdot|_\varepsilon$ is ε-Insensitive Loss Function defined by

$$|x|_\varepsilon = \begin{cases} 0 & |x| < \varepsilon \\ |x| - \varepsilon & \text{otherwise.} \end{cases}$$

In general, these minimization problems are not solved directly, and the following equivalent problem is solved:

Problem 2.1(nonlinear case)

$$\min_{f,\xi,\xi^*} \Phi(f,\xi,\xi^*) = \frac{C}{N} \sum_{i=1}^{N} (\xi + \xi^*) + \frac{1}{2}\|f\|^2 \qquad (2)$$

subject to constraints:

$$\begin{array}{rcll} f(x_i) - y_i & \leq & \varepsilon + \xi_1, & i = 1,\ldots,N \\ y_i - f(x_i) & \leq & \varepsilon + \xi_i^*, & i = 1,\ldots,N \\ \xi_i, \xi_i^* & \geq & 0 & i = 1,\ldots,N, \end{array} \qquad (3)$$

where new parameter C corresponds to $1/2\lambda$. Note that, the Problem 2.1 is a linearly constrained quadratic programming problem when the norm term in Eq.(2) is a quadratic function of the coefficients c, b. The above problem can be transformed to its dual problem by KKT condition. For detail, see [9]. The minimizer can be described as

$$f(x) = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(x_i + x) + b,$$

where α_i^* and α_i are the solution of the following dual problem:

Problem 2.2

$$\min_{\alpha^*,\alpha} \mathscr{W}(\alpha^*,\alpha) = \varepsilon \sum_{i=1}^{N} (\alpha_i^* + \alpha_i) - \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) + \frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i, x_j)$$

subject to the constraints:

$$\begin{array}{l} \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) = 0 \\ 0 \leq \alpha_i^*, \alpha_i \leq \frac{C}{N},\ i = 1,\ldots,N. \end{array}$$

The input vector x_i such that α_i^* or α_i are not equal to zero is called *support vector*. Any of the support vectors can be used to compute the parameter b. Actually, from the KKT condition, b can be given by

$$f(x_j) = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(x_i, x_j) + b = y_j + \varepsilon.$$

2.2 Stereo Matching and Similarity Map

In [10], stereo matching problem was translated to SVM regression problem as follows.

Let us consider the case of one-dimensional glayscale images. In the case of the two-dimensional images, if we know the epi-polar line, the problem reduce to the one-dimensional one.

Given two (left and right) images, set $l(x)$ and $r(x)$ as the intensity of the x-th pixel in the left and the right images, respectively. In addition we assume that the intensities are normalized to the range $[0:1]$. Suppose that an object is located in the x'-th pixel in the left image. If it is also located $(x'+a)$-th pixel in the right image, then the following relation should be hold:

$$l(x') - r(x'+a) \approx 0. \qquad (4)$$

Thus, the stereo matching problem can be formulated as follows: [1]

Find a function $a(x)$ in certain class which minimize the cost function

$$J_0 := \sum_x |l(x) - r(x + a(x))|. \qquad (5)$$

The function

$$\gamma(x,a) := 1 - |l(x) - r(x+a)| \qquad (6)$$

is called similarity function and can be used to measure the difficulty of this problem. In addition, the subset in the (x,a) space which consists of pairs with high similarity is called *similarity map*. In general, there exist many disparity a for given x such that the condition Eq.(4) holds, therefore, this problem is essentially ill-posed one.

In the regularization based approach[6], smoothing factor is added to J_0 such as

$$J_r := J_0 + \lambda \|f\|_P, \qquad (7)$$

where $\lambda > 0$ and is called regularization parameter. In Eq.(7), the norm $\|\cdot\|_P$ is certain measure of the smoothness of f. For example,

$$\int (d^{(k)}/dx^{(k)}) f(x) dx, k > 0$$

for some positive integer k may be used.

[1] In the case of color images, appropriate distance function in the color space (e.g., Euclid distance in RGB color space) may be used.

2.3 Stereo Matching as SVM Regression

The cost function (7) is not linear functional of f, thus we cannot directly apply the SVM regression approach to the stereo matching problem. However, if we consider select the pair (x,a) with high similarity as training example and if we choose x and a as input and output, the problem can be regarded as a kind of SVM regression problem. In [10], the problem was solved with the following steps:

1. Compute the similarity for all (x,a) pairs.
2. Select some (x,a) pairs with high similarity based on certain threshold value.
3. Miss-leading pairs are deleted by means of a *continuity-based* filtration.

In [10], a filter was applied in order to delete the isolated pairs (see Figure 1). In this case, gathered pairs can not

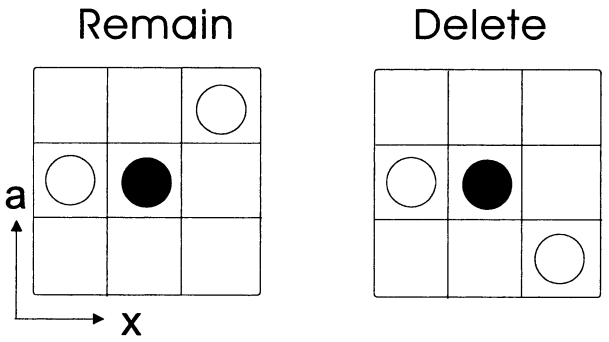


Figure 1: continuity based filtering

be made to be slim so that the number of the pair can not be decreased. Such cases can occur when monochromatic surface is seen.

An approaches to made the similarity map slim is that the filter is applied iteratively. However, in this case, some data corresponding the edge are also removed when right-and-left continuity is assumed. On the other hand, almost data remain when one-side continuity is assumed. Thus, this approach may leads unsatisfactory results.

To overcome this difficulty, we firstly extend the definition of similarity function. The similarity function is evaluated on the image window instead of single pixel, that is,

$$\gamma(x,a) = 1 - \frac{1}{2p+1}\sum_{i=-p}^{p}(l(x+p) - r(x+p-a)) \quad (8)$$

(One dimensional image).
This has the same effect as applying the filter in [10]. If we select single disparity which has highest similarity for each pixel, this is the equivalent to the well-known *block matching* method. However, when the size of the block is not large, this method causes the non-smooth results. So we applied the SVM regression approach to obtain smooth disparity function. This is summarized as follows:

1. Compute the **block**-similarity for all (x,a) pairs.
2. Select some (x,a) pairs with high similarity.

3 Experiments

The original images (Figure 2) are obtained by CCD video cameras (SONY EVI30D) and they are originally full color. Two cameras are located horizontally and parallel, so that the epi-polar line is horizontal and the positive disparity does not occurs. SVM regression problems are solved via mySVM [8] on IBM PC (Pentium III 933MHz, GCC 3.2.2). The size of the images are originally 640×320 and reduced to 80×60. As the color space, we use the CIE-Lab color space with Euclid distance. Any values are normalized to $[0:1]$. The range of the disparity is set to

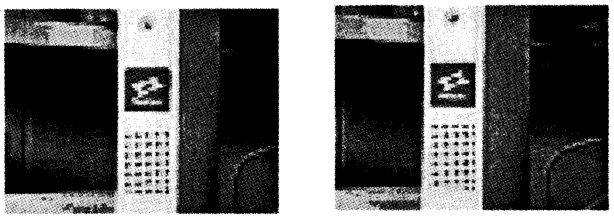


Figure 2: original images (left and right)

$[-15:5]$ and the block size is set to 5×5, so the rectangle area $(x,y) \in [18:73]\times[3:58]$ is used. The result of disparity perception based on the method in [10] is shown in Figure3. In this case, the number of the examples with high similarity ($\gamma > 0.99$) is 18550. In the disparity images, a pixel is blight when the disparity is small. We can see that the unsatisfactory result is obtained. Next, the data which have highest block similarities and are shown in Figure 4. The number of the examples with high block similarities is reduced to 8417. Figure 5 shows the disparity function obtained by applying the SVM regression. When applying the SVM regression, two examples that have highest similarity is used for each pixel. We can see that the satisfactory result is obtained rather than Figure 3 and Figure 4.

4 Conclusion

In this paper, the SVM regression approach is applied to the 3D reconstruction problem with real image. Using

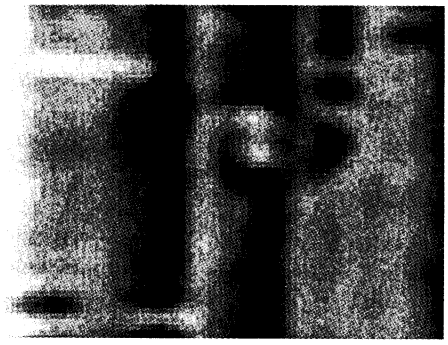


Figure 3: disparity image with single disparity and filtering

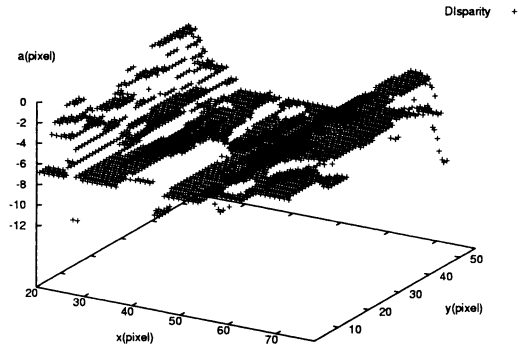


Figure 4: disparity function (by 5× 5 block matching based on 5 × 5 block similarity

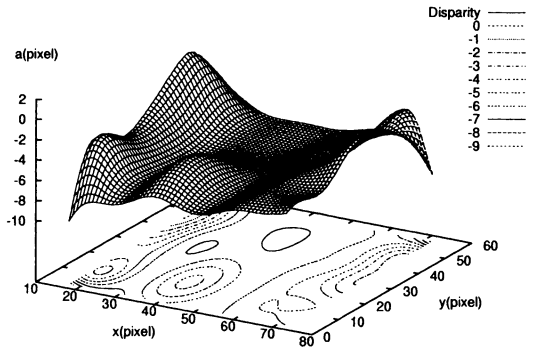


Figure 5: disparity function obtained by SVM regression

block similarity is shown to be effective to reduce the isolated examples. In the proposed approach, only the examples that posses highest similarity is used. We can also consider the SVM regression problem by considering (x, a) as input and γ as output. This approach will be examined in the future work.

References

[1] K.Achour and L.Mahiddine, "Hopfield neural network based stereo matching algorithm", **J. Mathematical Imaging and Vision** Vol. 16, 2002, pp.17–29.

[2] X.Hua, T.Mitsugi, Y.Tang, M.Yokomichi, and T.Kitazoe, "Stereo disparity perception for monochromatic flat slope based on neural net dynamical model", **Proc. 7th AROB**, 2002, pp.606–609.

[3] T.Joachims, "Text categorization with support vector machines: Learning with many relevant features", **Proc. Tenth European Conf. Machine Learning**, 1998, pp.137–142.

[4] D.Marr and T.Poggio, "Cooperative computation of stereo disparity", **Science** Vol. 194, 1976, pp.283–287.

[5] D.Marr and T.Poggio, "A computational theory of human stereo vision", *Proc. R. Soc. Lond.* B.204, 1979, pp.301–328.

[6] T.Poggio, V.Torre, and C.Koch, "Computational vision and regularization theory", **Nature** Vol. 317, No. 26, 1985, pp.314–320.

[7] M.Pontil and A.Verri, "Support vector machines for 3D object recognition", **IEEE Trans. PAMI**, Vol. 20, No. 6, 1998, pp.637–646.

[8] S. Rüping, **mySVM-Manual**, University of Dortmund, Lehrstuhl Informatik 8, http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/, 2000.

[9] V.N.Vapnik, **Statistical learning theory**, John Wiley and Sons, 1998.

[10] M.Yokomichi, H.Sugiyama, X.Hua and M.Kono, 3D-Perception via Support Vector Machine Regression, Proc. of SCI'03 (2003)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Edge detection and curvature calculation in the visual system

S. Niitsuma K. Tokunaga
sei@sys.eng.shizoka.ac.jp
Department of System Engineering, Faculty of Engineering, Shizuoka University
5-1, Johoku 3chome, Hamamatsu, 432-8561 Japan

Abstract

A theory of edge detection and calculating curvature is presented. It is assumed that the locations of receptive field centers of ganglion cells move with an input image. To determine them for input image, an optimization problem is fomalized. A solution of the problem gives an arrangement of receptive fields. An edge detection algorithm based on the arrangement is presented. Although the task of detecting edges has been reduced to that of finding the zero-crossing of $\nabla^2 G$ in the traditonal theory, it is reduced to the task of finding negative peaks which exist in the neighborhood of the zero-crossings. Edge detection algorithms are applied to the Helmholtz irradiation, a face image and a circele one. A clue to calculate curvature is found through an analysis of edges of a circular arc.

Keywords: edge detection, zero-crossing, ganglion cell, receptive field, endstopped cell

1 Introduction

The concept of an 'edge' has a visual and physical meaning. Some geometrical illusions originate in this dualism. We want to have a definition of an edge to be compatible with the illusions. In the traditional model of the early visual system, edge detection means to find the zero value of $\nabla^2 G * I$ for image I which is called the zero-crossings, where G is Gaussian distribution[1][2]. Fig.1 shows a well-known illusion called 'irradiation'. It cannot explain the illusion by having assumed the zero-crossing to be an edge(Fig.1(c)). On the other hand, if the edge is defined by a negative peak of $\nabla^2 G * I$ which is shown by a ridge in Fig.1(b), it can easily explain the illusion.

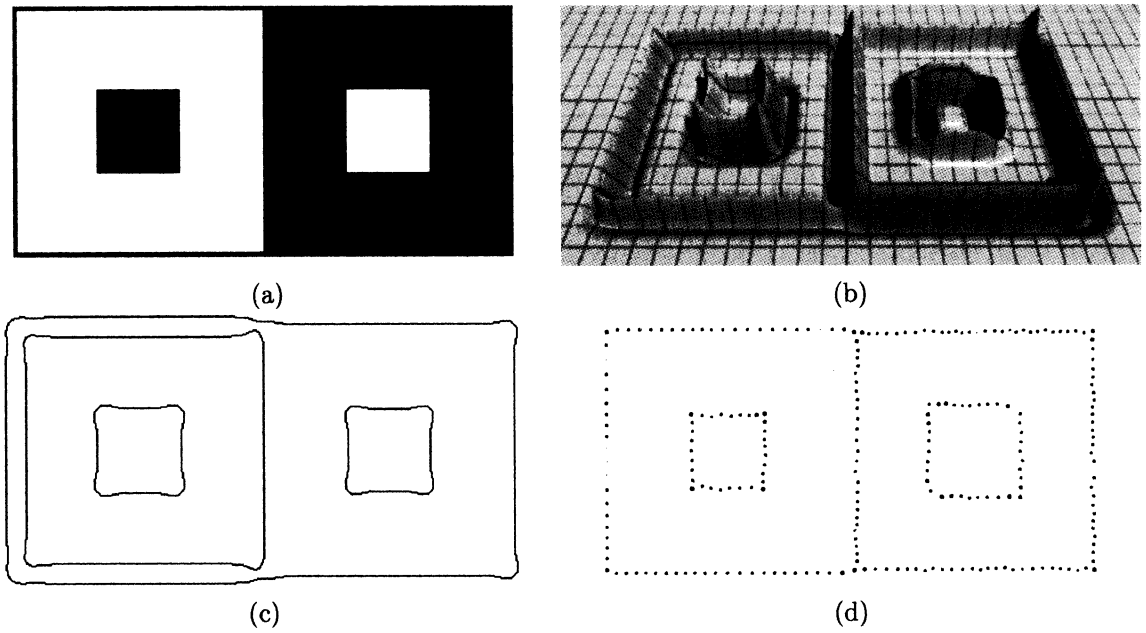


Fig.1: Explanation for the Helmholtz irradiation. This illusion is the one in which the two squares are identical in size but do not appear to be identical : the white square appears larger than the black. The image (a) has been convolved with $\nabla^2 G$ having $\sigma = 7.0[pixel]$. (b) and (c) show its 3-D representation and the zero-crossings, respectively. Note that we can find two squares with different size in (b) and the right square is larger than the left one. On the other hand, the two squares in (c) are identical in size. (d) The result of our edge-detection for the Helmholtz irradiation, where $R = 5.4[pixel]$

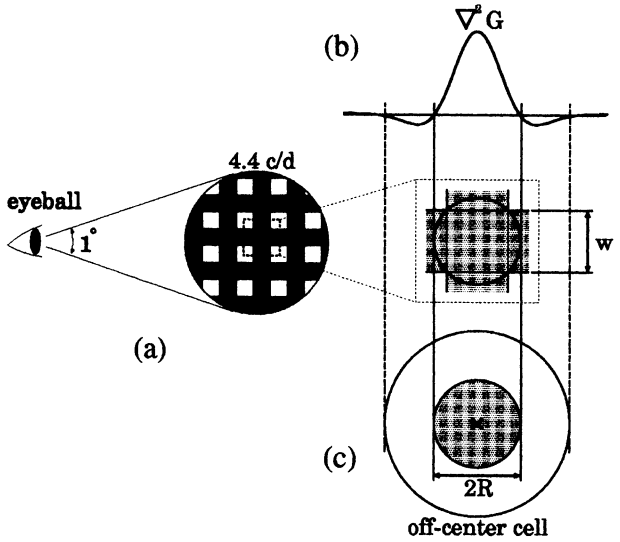


Fig.2: Relation among $\nabla^2 G$, a spatial frequency and a receptive field in the case of 4.4 cycle/deg.

2 Edge detection

How does the visual system find the negative peaks? This problem is solved with two kinds of ganglion cells, on-center cells and off-center cells. It is known that a ganglion cell has a receptive field described by $\nabla^2 G$, and that the positive and the negative part of $\nabla^2\ G * I$ are signaled by on-center cells and off-center cells, respectively. We base the detecting of the negative peaks by ganglion cells on the following assumptions:

(1) The size R of the excitatory region of the receptive field is decided from the spatial frequency which shows the spatial characteristics of the filter in the human retina(Fig.2). We adopt Wilson's six channels hypothesis [3].

(2) The center of a receptive field is not fixed but changes according to an input image.

Assumption (2) poses our next problem: how does one arrange the set of receptive fields for an input image? We formalize it as the following optimization problem: Let S be the set of visual cells. Suppose that each visual cell x_i has coordinates(x_{i^1}, x_{i^2}) and a density value $I\,(x_i)$, and the parameter of a partition is given by R that dnotes the size of the excitatory region. Then find the natural number C and a partition of S, $\{G_1, \ldots, G_c\}$, that minimize simultaneously both of the following criteria,

$$E = \max_{i=1,\ldots,c} \left| \max_{x_j \in G_i} \|x_j - v_i\| - R \right| \qquad (1)$$

and

$$J = \sum_{i=1,\ldots,c} \sum_{x_j \in G_i} \|x_j - v_i\|^2 \qquad (2)$$

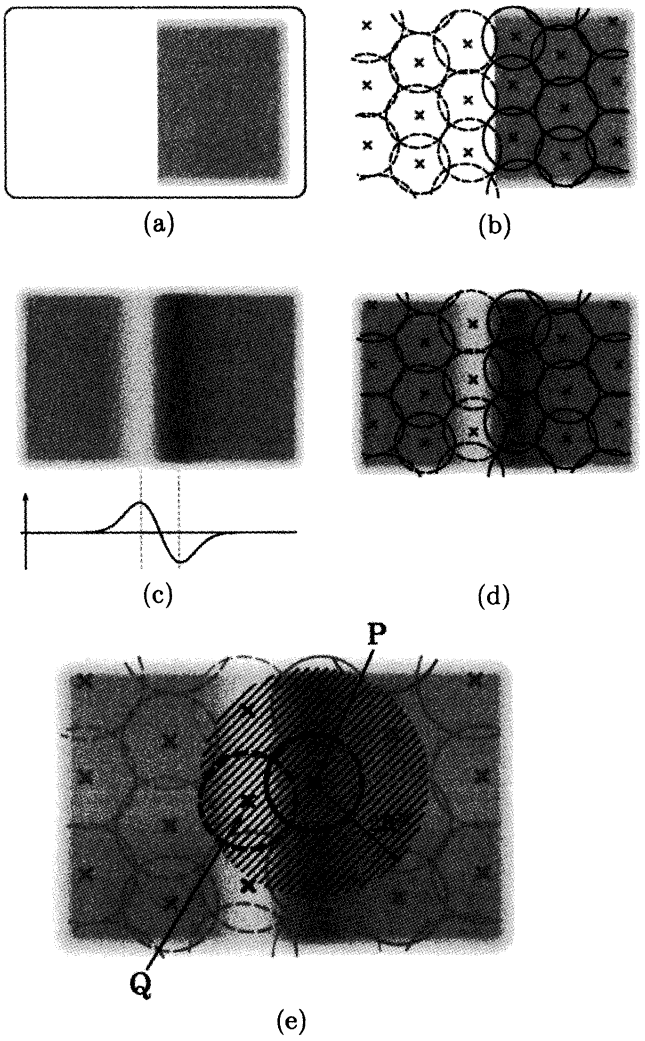
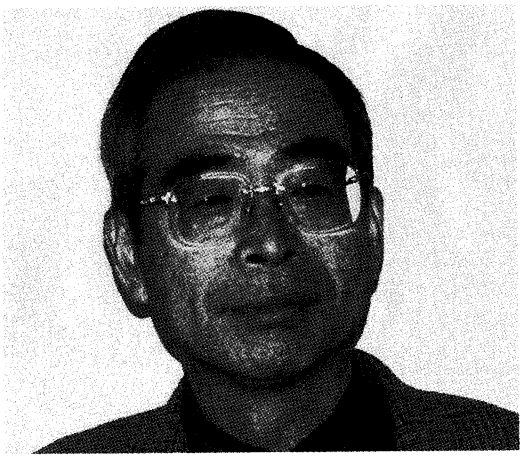
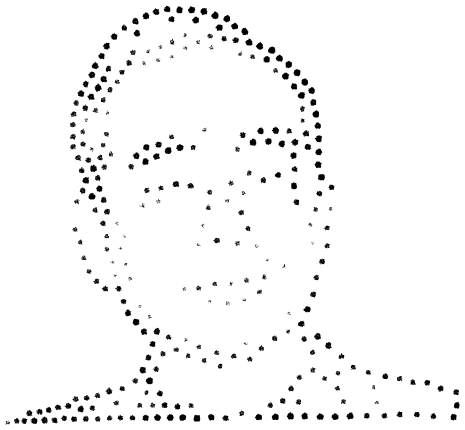


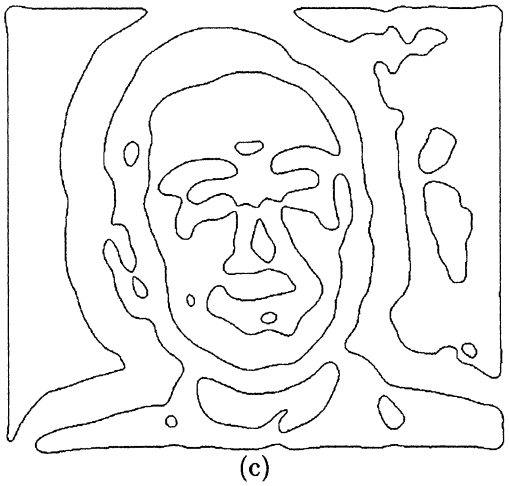
Fig.3: Arrangement of receptive fields and edge-detection. (b) shows an arrangement of the receptive fields for the image (a), where ⊛ and ⊛ denote the receptive field of off-center and on-center ganglion cell, respectively. (c) shows a 2-D representation of $\nabla^2\ G * I$ where I denotes the image. The negative peaks exist in a dark area. (d) is the result of superposing (b) on (c). It is possible to perceive the receptive fields of the off-center cells on the negative peaks. If these cells can know they are just on the negative peaks, they can detect the edge. (d) Judgement of an edge with off center cell. If P represents an off-center cell receptive field, and Q, an on-center one to be 2R or less distant from P, then if both are active, P is on the negative peak. We define an off-center cell to be active if the value of $\nabla^2 G$ at its center is negative, and an on-center cell is active if it is positive.



(a)



(b)



(c)

Fig.4: Comparison of edges of the negative peaks with the zero-crossings. The parameters are calculated from a spatial frequency 2.8[cycle/deg]. $R = 15.6[pixel]$ and $\sigma = 11.0[pixel]$.

where

$$v_i = \left(\sum_{x_j \in G_i} \frac{I(x_j) \times x_{j1}}{\sum_{x_j \in G_i} I(x_j)}, \sum_{x_j \in G_i} \frac{I(x_j) \times x_{j2}}{\sum_{x_j \in G_i} I(x_j)} \right) \quad (i = 1, \ldots, c).$$

A feasible solution for the optmization problem is obtained by clustering. The k-means which is a well-known clustering algorithm must be modified to solve it on the basis of R, since the k-means requires to be given the number of clusters. The modified k-means gives the set of coordinates of the receptive field centers(Fig.3(b)). If one perceives a topological relation between a negative peak and a positive one in the neighborhood of an edge, then one can find which off-center cell is on a negative peak(Fig.3(e)).

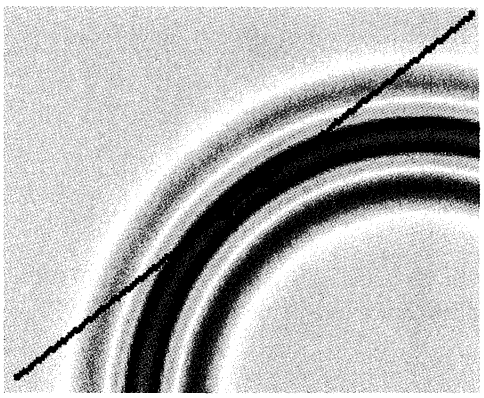
3 Detecting circular arcs

Fig.5(a) shows the covolution of a circle image with $\nabla^2 G$. One can perceive a change in $\nabla^2 G$ values along the line obtaind by connecting two edges. The change is different depending on a radius(Fig.5(b)). According to our theory, the distance between two edges is fixed even if the radius changes. Tne difference of the radius is reflected in the positions of positive peaks. Therefore, if one can calcurate the distance from the origin to a positive peak, then the radius of a circular arc is found (Fig.5(c)). The graph of $\nabla^2 G$ in Fig.5(b) is similar to the difference of two Gaussian distributions(DOG). This is reason why Dobbins' model is qualified for calculating carvature. Dobbins et al.[4] presented a mathematical model for endstopped cells in which the difference in response of two simple cells gives rise to endstopping and varies in proportion to curvature. Each receptive field of the simple cells has a Gaussian weighting along the long axis. Hence, the receptive field profile along the long axis of two simple cells is DOG.

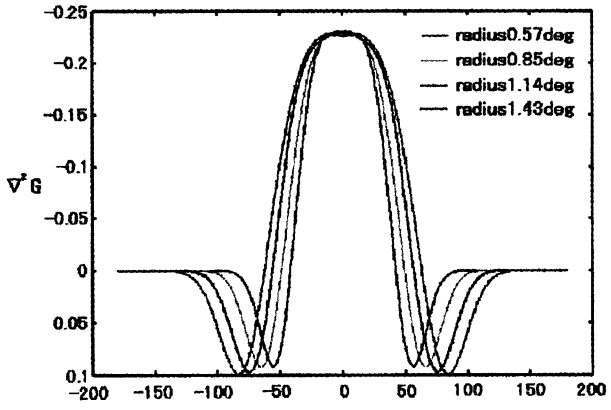
4 Conclusion

Our theory is based on an idea that a theory of the early visual system should be compatible with geometrical illusions. The idea led us to not zero-crossings but negative peaks as a definition of edges. The definition gives an explanation for the Helmholtz irradiation and natural edges for a face image. Endstopped cells are usually thought to be related to length or end-point detection. Dobbins et al. have suggested that these cells are able to calculate curvature of curves. Our results give a rationale for their model.

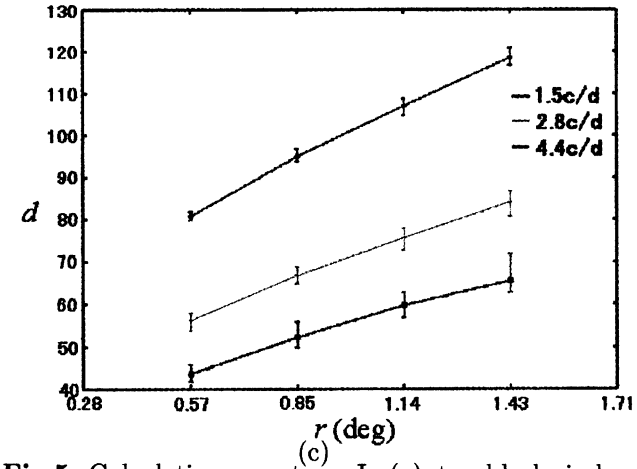
We thank T. Nakashima for programing support.



(a)



(b)



(c)

Fig.5: Calculating curvature. In (a), two black circles denote the edges of a circular arc image detected by our algorithm. The pattern of light and shade shows $\nabla^2 G$. We are perceived the change in $\nabla^2 G$ values along the line obtained by connecting two edges. (b) $\nabla^2 G$ values of four circles with refference to the distance from the origin which is settled at the center of two edges. Note that the position of a positive peak goes away from the origin as the radius increases. $R = 15.4[pixel]$, $\sigma = 11.0[pixel]$. (c) The relation between a radius(r) and a distance(d) from the origin to a positive peak. Each graph corresponds to one of spatial frequency 1.5, 2.8 4.4[cycle/deg]. This set of graphs poses our next problem : how does the visual system calculate the distance d ?

References

[1] D. MARR, "VISION, A Computational Investigation into the Human Representation and Processing of Visual Information." Information, W.H. Freeman, New York, 1982.

[2] D. MARR, and E. HILDRETH, " Theory of edge detection." Proc. R. Soc. Lond. B, vol.207, pp.187-217, 1980. Information, W.H. Freeman, San Francisco, 1982.

[3] H.R. Wilson, D.K. McFarlane, and G.C. Phillips, " Spatial frequency tuning of orientation selective units estimated by oblique masking, " Vision Research, vol.23, no.9, pp.878-882, 1983.

[4] A. Dobbins, S.W. Zucker and M.S. Cynader, "End-stopped neurons in the visual cortex as a substrate for calculating curvature." Nature, vol.329, pp.438-441, 1987

[5] Hubel, D.H. and Wiesel, T.N. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex" J. Physiol., vol.160, pp.106-154, 1962

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Three-Dimensional Multi-Inkdot Automata

Tadayuki Makino †, Hidenobu Okabe †, Shinya Taniguchi †,
Makoto Sakamoto †, and Katsushi Inoue ‡

†Dept of Computer Science and Systems Engineering, Miyazaki University, Miyazaki 889-2192, JAPAN
‡Dept of Computer Science and Systems Engineering, Yamaguchi University, Ube 755-8611, JAPAN

Abstract

Recently, related to the open problem whether deterministic and nondeterministic space (especially lower-lebel) complexity classes are separated, inkdot Turing machines were introduced. On the other hand, due to the advances in many application areas about three-dimensional information processing, the research of three-dimensional automata as the computational of three-dimensional pattern processing has been meaningful. From this viewpoint, we propose a three-dimensional multi-inkdot automata, and investigate some properties of three-dimensional multi-inkdot automata.
keywords: inkdot automaton, marker automaton, alternation, three-dimensional data processing, computational complexity.

1 Introduction and Preliminaries

As is the well-known open problems in computational complexity, there is the historical open question whether or not the separation exists between deterministic and nondeterministic space (especially lower-level) complexity classes. Related to the open question, Ranjan, Chang, and Hartmanis introduced a slightly modified Turing machine model, calld a one-inkdot Turing machine [9]. An inkdot machine is a conventional Turing machine capable of dropping an inkdot on a given input tape for landmark, but unable to further pick it up [3-9]. Against an earlier expectation, it was proved that nondeterministic inkdot Turing machines are more powerful than nondeterministic ordinary Turing machines for sublogarithmic space bounds. As is well-known result in the case of two-dimensional input tapes [1], there is a set of square tapes accepted by a nondeterministic finite automaton but not by any deterministic Turing machine with sublogarithmic space bounds. Thus, it makes no sense to ask the same question whether the separation exists between deterministic and nondeterministic complexity classes for the two-dimensional Turing machines. On the other hand, there is another important aspect in the inkdot mechanism: we can see a two-dimensional finite automaton with inkdots as a weak recognizer of the inherent properties of digital pictures. By this motivation, a finite state version of the inkdot machine was introduced [8]. By the way, the question of whether processing three-dimensional digital patterns is much more difficult than two-dimensinal ones is of great interest from the theoretical and practical standpoints. Thus, the research of three-dimensional automata as the computational model of three-dimensional pattern processing has been meaningful [11]. From this viewpoint, we introduce a three-dimensional multi-inkdot automaton, and investigate some properties of three-dimensional multi-inkdot automata in this paper.

Let Σ be a finite set of symbols. A three-dimensional tape over Σ is a three-dimensional rectangular array of elements of Σ. The set of all the *three-dimensional imput tapes* over Σ is denoted by $\Sigma^{(3)}$. Given a tape $x \in \Sigma^{(3)}$, for each

$j(1 \leq j \leq 3)$, we let $l_j(x)$ be the length of x along the jth axis. When $1 \leq x_j \leq l_j(x)$ for each $j(1 \leq j \leq 3)$, let $x(x_1, x_2, x_3)$ denote the symbol in x with coordinates (x_1, x_2, x_3), as shown in Figure 1.

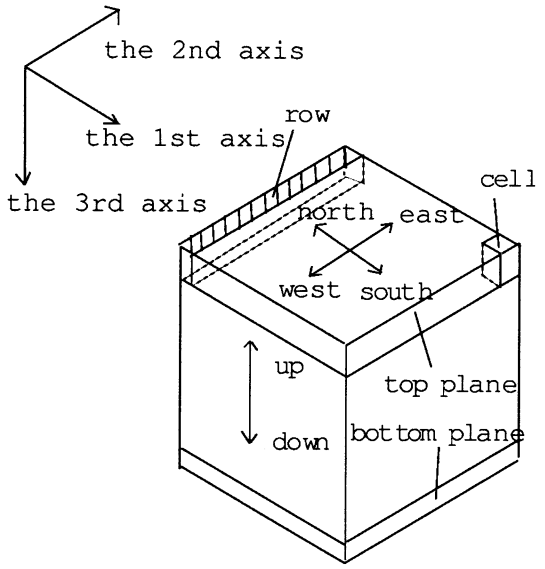


Figure 1: Three-dimensional input tape.

A *three-dimensional alternating finite automaton* (3-AF) M is a *three-dimensional alternating Turing machine* with no workspace whose state set is partitioned into *universal* and *existential* states [2,10]. The machine M has a read-only three-dimensional tape. A *step* of M consists of reading one symbol from each tape, moving the input head in specified direction $d \in \{east, west, south, north, up, down, no\ move\}$, and entering a new state, in accordance with the *next-move relation*. A *three-dimensional alternating k-inkdot automaton* (3-AI_k), $k \geq 1$, is a 3-AF with k dots of ink. Thus a 3-AI_k is a 3-AF capable to drop an inkdot at most k tape-cell on the input for a landmark, once on each cell, but not to pick it up nor erase it any more [9]. In order to distinguish among determinism, nondeterminism, alternation with only universal states, and alternation, we denote a deterministic version of 3-AF [nondeterministic version of 3-AF, alternating version of 3-AF with only universal states, determinisitic version of 3-AI_k, nondeterministic version of 3-AI_k, alternating version of 3-AI_k with only universal states] by 3-DF [3-NF, 3-UF, 3-DI_k, 3-NI_k, 3-UI_k], respectively. Let $\pounds$[3-AF] = $\{T|T$ is accepted by some 3-$AF\}$. $\pounds$[3-AI_k], etc. are defined similarly. For any family of three-dimensional automata A's, $\pounds[A^c]$ denotes the class of sets of cubic tapes accepted by A's. For a set T of three-dimensional tapes, the complementation of T is denoted by $\hat{T}$. Define co-$\pounds = \{\hat{T}|T \in \pounds\}$.

2 Results

We first investigate the properties of deterministic version of inkdot automata, and show that deterministic inkdot automata are equivalent to ordinary deterministic finite automata. That is, no hierarchy exists based on the number of inkdots for deterministic case.

Theorem 2.1. $\pounds$[3-DF] = $U_{k\geq 1}\pounds$[3-DI_k].

Proof: The simulation method of Ref.[9], which only treats one-ink machines, is also valid in our three-dimensional case. We recall this technique for the beginning: 3-DF M' behaves in the same way as 3-DI_1 M until it uses its own ink. When M will drop its ink, M' memorizes the input symbol on the current cell without ink drop. After that, M' continue to simulate M, except when M' encounters the same input symbol as in the finite control. When this happens, M' memorizes the current state of M and performs *depth-first backward search* [9] of the computation of M toward the initial configuration to test whether the encountered cell was the place of ink drop. If M' reaches the initial configuration of M, it then performs *forward simulation* of M until it will drop the ink. After that, M' continue to simulate M from the memorized state as if there is an ink at this position. Based on the idea above, we will prove the theorem by induction on the number of inkdots. That is, assuming that 3-DF's can simulate 3- DI'_ks, we will show that a 3-DI_k M' can

simulate the given 3-DI_{k+1} M. Before M drops the $k+1$st ink, when each time M' encounters any of the previously dropped inks, it memorize the pair of the ink name and the current state to use just like the initial configuration in the case of $k=1$ described above. If M would drop the $k+1$st ink, M' memorizes the current input symbol in the finite control. After that, M' continues to simulate M, except when M' encounters the same input symbol as the recorded symbol in the finite control. In this case, M' performs depth-first backward search and forward simulation of M between the current configuration and the most recent configuration that encountered the inkdots represented by the newest pain in the finite control. □

Next, we investigate the properties of nondeterministic version of inkdot automata, and show that nondeterministic 1-inkdot automata are more powerful than ordinary nondeterministic finite automata.

Lemma 2.1. *Let* $T_1 = \{x \in \{0,1\}^{(3)} \mid {}^{\exists}m \geq 1 [l_1(x) = l_2(x) = l_3(x) = 2m \ \& \ \forall i, \forall j, \forall k \ (1 \leq i \leq 2m, 1 \leq j \leq 2m, 1 \leq k \leq m)[x(i,j,k) = x(i,j,k+m)]]\}$. *Then* (1) $\hat{T}_1 \in \pounds[3\text{-}NI_1^c]$, (2) $\hat{T}_1 \notin \pounds[3\text{-}AF^c]$.
Proof: (1) a 3-NI_1 M accepting T_1 acts as follows. Given an input tape x with $l_1(x) = l_2(x) = l_3(x) = 2m$, M firstly scans the kth plane from the northmost row to the southmost row in a plane and from the westmost cell to the eastmost cell in a row. This sequence of moves is called a *systematic scan*. M guesses some position (i,j,k) such that $x(i,j,k) \neq x(i,j,k+m)$ in the systematic scan. Then, it drops the ink at (i,j,k) and goes to the southeastmost boundary of the kth plane, i.e., the position $(2m+1, 2m+1, k)$ in the systematic scan. Next, M moves its input head one cell down for every two north moves and two west moves of input head until M reaches the northwestmost boundary of the $(k+m)$th plane, i.e., the position $(0,0,k+m)$. Then, while going to the southeastmost boundary of the $(k+m)$th plane in the systematic scan, it existentially branches two machines, one of which continues to go to the southeastmost boundary, and the other of which memorizes the input symbol 'b' on the current position and goes upward to check the inequality $b \neq (i,j,k)$. If this check succeeds, then it enters an accepting state. Otherwise, it halts in a non-accepting state. It is clear that $T(M) = T_1$. (2): By using the same idea in the proof of part (2) of Lemma 5.1 in [8], we can easily show that $\hat{T}_1$ is not in $\pounds[3\text{-}AF^c]$. □

From Lemma 2.1 and the trivial fact $\pounds[3\text{-}NF] \subseteq \pounds[3\text{-}AF]$, we can get

Theorem 2.2. $\pounds[3\text{-}NF^c] \subsetneq \pounds[3\text{-}NI_1^c]$.

We next investigate the properties of alternating inkdot automata which have only universal states.

Theorem 2.3. $\pounds[3\text{-}UF^c] \subsetneq \pounds[3\text{-}UI_1^c]$.
Proof: In the proof of Part (1) of Lemma 2.1, the constructed 3-NI_1 M accepting $\hat{T}_1$ always halts, i.e., it never enters a loop for any time for any input. Based on this fact and by exchanging the existential states of M to universal states, non-accepting states to accepting states, and accepting states to non-accepting halting states, we can obtain a 3-UI_1 accepting T_1. On the other hand, from the trivial fact $\pounds[3\text{-}UF] \subseteq \pounds[3\text{-}AF]$ and Part (2) of Lemma 2.1, we get the desired result $T_1 \notin \pounds[3\text{-}UF^c]$. □

Theorem 2.4. co-$\pounds[3\text{-}UI_k] \subseteq \pounds[3\text{-}NI_{k+1}]$, *for each* $k \geq 0$.
Proof: Without loss of generality, we assume that, when a 3-UI_k M wants to reject a given input x, it always enters a loop. From this assumption, we can say that M does not accept x iff there exists a computation path $P_M(x)$ such that $c_0 \vdash_M c_1 \vdash_M \cdots c \vdash_M \cdots \vdash_M c$, where c_0 is the initial configuration and the cycle $c \vdash_M \cdots \vdash_M c$ represents a loop. (Given a M, we write $c \vdash_M c'$ and say c' is a *successor* of c if configuration c' follows from configuration c in one step of M, according to the *transaction rules*.) To simulate the complement of M, a 3-NI_{k+1} guesses $P_M(x)$ by using its existential states and halts in a non-accepting state when M will enter an accepting state. M' uses k inks for the trace of the prefix subpath

$c_0 \vdash_M \cdot \vdash_M c$ in which M consumes at most k inks. The $k+1$st ink of M' is used for the detection of the loop $c \vdash_M \cdots \vdash_M c$. Note that M' never enters an accepting state if any computation of M has no loop. □

Corollary 2.1. co-$\mathcal{L}$[3-UF] $\subseteq \mathcal{L}$[3-NI_1].

Finally, we investigate the properties of alternating inkdot automata which can use both universal states and existential states of finite control. We have

Theorem 2.5. co-$\mathcal{L}$[3-AI_k] $\subseteq \mathcal{L}$[3-AI_{k+1}], *for each* $k \geq 0$.

Proof: Without loss of generality, we assume that, when a 3-AI_k M rejects the given input x, it always enters a loop. From this, we can say that M does not accept x iff there exists a non-accepting computation tree which has no leaf, i.e. any path descendant from the root of which enters an infinite loop. At each step of M, a 3-AI_{k+1} M' existentially branches into two machines, one of which drops an ink for detecting a loop starting from the current configuration of M. Both machines then continue further simulation of M under the conditions such that the universal and existential states of M are exchanged with each other in M'. Of course, when M halts in an accepting state, M' halts in a non-accepting state. If there exists a non-accepting (infinite) computation tree of M, then there exists an accepting finite computation tree of M'. Conversely, if there exists an accepting computation (finite) tree of M, then there exists a non-accepting finite computation tree of M'. Note that M' uses k inks until M enters a loop and uses one ink to verify the existence of a loop. □

Corollary 2.2. co-$\mathcal{L}$[3-NF] $\cup$ co-$\mathcal{L}$[3-UF] $\subseteq \mathcal{L}$[3-AI_1].

3 Conclusion

In this paper, we proposed a three-dimensional multi-inkdot automaton as a new type of computational model of three-dimensional pattern processing, and investigated some accepting powers of this automaton. It will be interesting to investigate about recognizability of connected pictures by three-dimensional multi-inkdot automata.

References

[1] M.Blum and C.Hewitt Automata on a two-dimensional tape, *in IEEE Symposium on Switching and Automata Theory*, pages155-160,1967.

[2] A.K.Chandra, D.C.Kozen, and L.J.Stockmeyer, Alternation, *Journal of the Association for Computing Machinery*, 28 (1):114-133,1981.

[3] V.Geffert, Nondeterministic computations in sublogarithmic space and space constructibility, *SIAM Journal on Computing*, 20 (3):484-498,1991.

[4] K.Inoue and A.Nakamura, An n-dimensional on-line tessellation acceptor (in Japanese), *The Transactions of the IECE*, J59-D(4):229-236,1976.

[5] K.Inoue, A.Ito, and I.Takanami, A relationship between nondeterministic Turing machines and 1-inkdot Turing machines with small space, *Information Processing Letters*, 43:225-227,1992.

[6] K.Inoue, A.Ito, I.Takanami, and T.Yoshinaga, A note on multi-inkdot nondeterministic Turing machines with small space, *Information Processing Letters*, 48:285-288,1993.

[7] K.Inoue, A.Ito, and I.Takanami, On 1-inkdot alternating Turing machines with small space, *Theoretical Computer Science*, 127:171-179,1993.

[8] A.Ito, K.Inoue, I.Takanami, and Y.Wang, The effect of inkdots for two-dimensional automata, *Internatinal Journal of Pattern Recognition and Artificial Intelligence*, 9(5):777-796,1995.

[9] D.Ranjan, R.Chang, and J.Hartmanis, Space bounded computations : review and new separation results, *Theoretical Computer Science*, 80:289-302,1991.

[10] M.Sakamoto, K.Inoue, and I.Takanami, A note on three-dimensional alternating Turing machines with space smaller than $\log m$, *Information Sciences*, 72:225-249,1993.

[11] M.Sakamoto, Three-dimensional alternating Turing machines,*Ph. D. Thesis*,1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Stabilization of LTI Systems with Periodic Communication Constraints by using Periodic Kalman Filter

Michio Kono
Faculty of Engineering
Miyazaki University,
1-1 Gakuenkibanadai Nishi,
Miyazaki, Japan, 889-2192

Atsunori Kose
Graduate School of Engineering
Miyazaki University,
1-1 Gakuenkibanadai Nishi,
Miyazaki, Japan, 889-2192

Nobuya Takahashi
Faculty of Engineering
Miyazaki University,
1-1 Gakuenkibanadai Nishi,
Miyazaki, Japan, 889-2192

Abstract

This paper considers feedback control systems wherein the control loops are closed through a real time network, and expresses the LTI system with the constraint in an input/output as a periodic discrete time system. We shall stabilize this system by using periodic Kalman filter. A numerical example and simulation result show that our method is effective.

1 Introduction

For the problem of remote control system in which sensors and actuators are connected via network, it is important how to transmit signals under constrained capacity of communication bus. If the bus capacity is small, it cannot transfer all input/output signals at one time and causes bad influence on system performance. In [2], [3], [4], it is shown that LTI systems with periodic communication constraints are described by periodic discrete-time systems with the special structure and stabilized using static output feedback, observer based controller or output sample hold control. We shall stabilize this system by periodic Kalman filter.

2 Preliminaries

2.1 Description of the networked control system

Let the plant be an LTI (linear time-invariant) discrete time system

$$\begin{aligned} x(t+1) &= Ax(t)+Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $x \in \Re^n$ is a state vector, $u \in \Re^m$ is a actuator-input vector and $y \in \Re^p$ is a sensor-output vector. The controller is connected to system (1) via network. Because of the constraints on capacity of a communicating bus, only partial signals of the controller-output are transmitted to the plant. Only the corresponding signals of the actuator-input are updated and other signals are held until next signals arrive. Similarly, only partial signals of the sensor-output are transmitted to the controller.

We assume that the controller-to-plant communication follows ω-periodic pattern. Let a controller-actuator update pattern denote an ω-periodic vector $\sigma_c(t) \in \{0,1\}^m$ where the i-th element of $\sigma_c(t)$ is 1 if the i-th input is updated at time t and is 0 otherwise. For a sensor-controller update pattern, an ω-periodic vector $\sigma_o(t) \in \{0,1\}^p$ is defined similarly. $\sigma_c(t)$ and $\sigma_o(t)(t = 0,1,2,\ldots)$ are called communication sequences. Using $\sigma_c(t)$, $D_c(\cdot)$ is defined as

$$\begin{aligned} D_c(t,t) &= \mathrm{diag}(\sigma_c(t)) \\ D_c(t,s) &= (\mathbf{I}_m - \mathrm{diag}(\sigma_c(t)))(\mathbf{I}_m - \mathrm{diag}(\sigma_c(t-1)))\cdots \\ &\quad \cdot(\mathbf{I}_m - \mathrm{diag}(\sigma_c(t-s+1)))\mathrm{diag}(\sigma_c(s)) \\ &\qquad s = t-1,\ldots,t-\omega+1 \end{aligned} \tag{2}$$

where $\mathrm{diag}(x)$ denotes the diagonal matrix whose diagonal elements are x_i. And, $\mathbf{I}_i$ denotes i-dimensional identity matrix. Similarly, using $\sigma_o(t)$, $D_o(\cdot)$ is defined as

$$\begin{aligned} D_o(t,t) &= \mathrm{diag}(\sigma_o(t)) \\ D_o(t,s) &= (\mathbf{I}_p - \mathrm{diag}(\sigma_o(t)))(\mathbf{I}_p - \mathrm{diag}(\sigma_o(t-1)))\cdots \\ &\quad \cdot(\mathbf{I}_p - \mathrm{diag}(\sigma_o(t-s+1)))\mathrm{diag}(\sigma_o(s)) \\ &\qquad s = t-1,\ldots,t-\omega+1 \end{aligned} \tag{3}$$

Then, the networked control system is expressed as ω-periodic system.

$$\begin{aligned} x_e(t+1) &= A_e(t)x_e(t)+B_e(t)u_\iota(t) \\ y_\iota(t) &= C_e(t)x_e(t) \end{aligned} \tag{4}$$

where y_t is controller-input signal and u_t is controller-output signal. This system has an extended state vector.

$$x_e(t) := \left[\begin{array}{c} x(t) \\ x(t-1) \\ \vdots \\ x(t-\omega+1) \\ u_t(t-1) \\ u_t(t-2) \\ \vdots \\ u_t(t-\omega+1) \end{array}\right] \begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} n\omega \\ \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} m(\omega-1) \end{array}$$

$$n_e := \omega(m+n)-m,$$

$A_e(t), B_e(t)$ and $C_e(t)$ are defined as follows [4].

$$A_e(t) = \left[\begin{array}{cc} A_{e_{11}} & A_{e_{12}}(t) \\ \mathbf{O} & A_{e_{22}} \end{array}\right] \begin{array}{l} \\ \left.\begin{array}{c} \\ \end{array}\right\} m(\omega-1) \end{array} \quad (5)$$

$$A_{e_{11}} = \left[\begin{array}{cccc} A & \mathbf{O}_{n\times n} & \cdots & \mathbf{O}_{n\times n} \\ \mathbf{I}_n & & & \vdots \\ \vdots & \ddots & & \vdots \\ \mathbf{O}_{n\times n} & \cdots & \mathbf{I}_n & \mathbf{O}_{n\times n} \end{array}\right]$$

$$A_{e_{12}}(t) = \left[\begin{array}{ccc} BD_c(t,t-1) & \ldots, & BD_c(t,t-\omega+1) \\ \hline \multicolumn{3}{c}{\mathbf{O}_{n(\omega-1)\times m(\omega-1)}} \end{array}\right]$$

$$A_{e_{22}} = \left[\begin{array}{cccc} \mathbf{O}_{m\times m} & \cdots & \cdots & \mathbf{O}_{m\times m} \\ \mathbf{I}_m & & & \vdots \\ & \ddots & & \vdots \\ \mathbf{O}_{m\times m} & & \mathbf{I}_m & \mathbf{O}_{m\times m} \end{array}\right]$$

$$B_e(t) = \left[\begin{array}{c} BD_c(t,t) \\ \mathbf{O}_{(n-1)\omega\times m} \\ \mathbf{I}_m \\ \mathbf{O}_{m(\omega-1)\times m} \end{array}\right] \begin{array}{l} \left.\begin{array}{c} \\ \end{array}\right\} n \\ \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} n\omega \end{array} \quad (6)$$

$$\begin{aligned} C_e(t) &= [D_o(t,t)C, \ldots \\ &\quad , D_o(t,t-\omega+1)C, \mathbf{O}_{p\times m(\omega-1)}] \end{aligned} \quad (7)$$

where $\mathbf{O}_{i\times j}$ denotes $i \times j$ zero matrix.

Let we define the transition matrix

$$\begin{aligned} X_e(t_0) &= X_e(t_0, t_0-\omega) \\ &= A_e(t_0-1)\cdots A_e(t_0-\omega) \\ &= \left[\begin{array}{cc} X_{e_{11}} & X_{e_{12}}(t_0) \\ \mathbf{O} & X_{e_{22}} \end{array}\right] \end{aligned} \quad (8)$$

Then, it follows from (5) that

$$X_{e_{11}} = A^{\omega}_{e_{11}} \quad (9)$$

$$X_{e_{22}} = A^{\omega}_{e_{22}} = 0 \quad (10)$$

$$\begin{aligned} X_{e_{12}}(t_0) &= A^{\omega-1}_{e_{11}} A_{e_{12}}(t_0-\omega) \\ &\quad + A^{\omega-2}_{e_{11}} A_{e_{12}}(t_0-\omega+1) A_{e_{22}} \\ &\quad + A^{\omega-3}_{e_{11}} A_{e_{12}}(t_0-\omega+2) A^2_{e_{22}} \\ &\quad \cdots + A_{e_{11}} A_{e_{12}}(t_0-2) A^{\omega-2}_{e_{22}} \end{aligned} \quad (11)$$

Remark 1 *$X_e(t_0)$ is called a monodromy matrix of the system (4) at time t_0 and its eigenvalues are independent of t_0.*

Remark 2 *Stability of the system (4) is dominated by the eigenvalues of monodromy matrix $X_e(t)$.*

The following proposition shows a special property of the monodromy matrix of the system (4).

Proposition 1 *$X_e(t_0)$ has $(m+n)(\omega-1)$ zero eigenvalues.*

Now we define

$$\begin{aligned} Q_e(t_0) &= [X_e(t_0, t_0-\omega+1)B_e(t_0-\omega), \\ &\quad X_e(t_0, t_0-\omega+2)B_e(t_0-\omega+1), \\ &\quad \ldots, B_e(t_0-1)] \end{aligned} \quad (12)$$

Then, we have the following result:

Theorem 1 *[5] The system (4) is reachable at t_0 if and only if*

$$\begin{aligned} &rank[Q_e(t_0), X_e(t_0)Q_e(t_0), \ldots, X^{n_e-1}_e(t_0)Q_e(t_0)] \\ &= n_e \end{aligned} \quad (13)$$

2.2 Stabilization by periodic state feedback

If the ω-periodic system (4) is stabilizable, it can be decomposed into the reachable subsystem and the unreachable one. Since for the reachable subsystem eigenvalues of the monodromy matrix can be assigned by periodic state feedback[5], the whole system is stabilized by periodic state feedback

$$\begin{aligned} u_t(t) &= F_e(t)x_e(t) \\ F_e(t) &= F_e(t+\omega) \end{aligned} \quad (14)$$

However, since the state is not available in the networked control system, we need a state estimator. In the next section, we introduce the periodic Kalman filter.

3 Periodic Kalman filter

Consider the following model which includes the system noise $v(t)$ and observation noise $w(t)$.

$$\begin{aligned} x_e(t+1) &= A_e(t)x_e(t) + B_e(t)u_\iota(t) + v(t) \\ y_\iota(t) &= C_e(t)x_e(t) + w(t), \end{aligned} \tag{15}$$

where $\{v(t)\}$ and $\{w(t)\}$ are uncorrelated zero mean random sequences satisfying

$$E\{v(s)v^T(t)\} = V(t)\delta_{s,t} \tag{16}$$
$$V(t) \geq 0, \quad V(t+\omega) = V(t)$$
$$E\{w(s)w^T(t)\} = W(t)\delta_{s,t} \tag{17}$$
$$W(t) \geq 0, \quad W(t+\omega) = W(t)$$

with $\delta_{s,t}$ being the Kronecker delta. The initial state $x(0)$ is assumed to be a random variable uncorrelated with $\{v(t)\}$ and $\{w(t)\}$ having mean $\bar{x}_0$ and covariance matrix $\sum_o$. When $y_\iota(0), \ldots, y_\iota(t)$ are obtained, the least square estimate $\hat{x}_e(t)$ is given as follows

$$\begin{aligned} \hat{x}_e(t) &= \tilde{x}_e(t) + K(t)[y_\iota(t) - (C_e(t)\tilde{x}_e(t) + \bar{w}(t))] \\ \tilde{x}_e(t) &= A_e(t-1)\tilde{x}_e(t-1) \\ &\quad + B_e(t-1)u_\iota(t-1) + \bar{v}(t-1) \\ \hat{x}_e(0) &= \bar{x}_e(0) \end{aligned} \tag{18}$$

where

$$K(t) = \Sigma(t)C_e^T(t)[C_e(t)\Sigma(t)C_e^T(t) + W(t)]^{-1} \tag{19}$$

$\Sigma(t)$ is the stationary periodic solution of the periodic Riccati difference equation.

$$\begin{aligned} \Sigma(t) &= A_e(t-1)\Sigma(t-1)A_e^T(t-1) - A_e(t-1) \\ &\quad \cdot\Sigma(t-1)C_e^T(t-1)[C_e(t-1)\Sigma(t-1)C_e^T(t-1) \\ &\quad + W(t-1)]^{-1}C_e(t-1)\Sigma(t-1)A_e^T(t-1) \\ &\quad + V(t-1) \end{aligned} \tag{20}$$

This estimator is called periodic Kalman filter.

Remark 3 *The condition for the stationary periodic solutions to exist is given in [6].*

4 Numerical example

Let consider the example treated in reference [2],[3],[4].

$$x(k+1) = \begin{bmatrix} 1 & 3/4 & 1/2 & 0 \\ 1/4 & 3 & 1/3 & -1/3 \\ 1/6 & 0 & -1/2 & -3/7 \\ 0 & -1 & 2/5 & 0 \end{bmatrix} x(k)$$
$$+ \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} u(t) + v(t) \tag{21}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(t) + w(t) \tag{22}$$

$$W = 0.01 \cdot \mathbf{I}_2$$
$$V = \begin{bmatrix} \mathbf{I}_4 & \mathbf{O}_{6\times4} \\ \mathbf{O}_{6\times4} & \mathbf{O}_{6\times6} \end{bmatrix}$$

For this system, we choose $\omega = 2, t_0 = 0$. Then we get the periodic system with a monodromy matrix

$$X_e = \left[\begin{array}{cccc|c} 1.2708 & 3.0000 & 0.5000 & -0.4613 & \\ 1.0556 & 9.2508 & 0.8250 & -1.1429 & \\ 0.0833 & 0.5536 & 0.1619 & 0.2143 & \\ -0.1833 & -3.0000 & -0.5222 & 0.1619 & \\ 1.0000 & 0.7500 & -0.5000 & 0.0000 & \\ 0.2500 & 3.0000 & 0.3333 & -0.3333 & \\ 0.1667 & 0.0000 & -0.5000 & -0.4286 & \\ 0.0000 & -1.0000 & 0.4000 & 0.0000 & \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & \end{array}\right.$$
$$\left.\begin{array}{c|c} 0.7500 & \\ 3.0000 & \\ 0.0000 & \\ -1.0000 & \mathbf{O}_{6\times1} \\ 0.0000 & \\ 1.0000 & \\ \hline & \\ \mathbf{O}_{4\times6} & \end{array}\right] \tag{23}$$

Let $Y_e(2,0)$ be a monodromy matrix of the closed-loop system

$$x_e(t+1) = (A_e(t) + B_e(t)F_e(t))x_e(t), \tag{24}$$

that is,

$$Y_e(2,0) = (A_e(1) + B_e(1)F_e(1))(A_e(0) + B_e(0)F_e(0)) \tag{25}$$

Characteristic of the system (24) is dominated by eigenvalues of $Y_e(2,0)$. Choosing $F_e(1) = 0$ yields

$$Y_e(2,0) = X_e(0) + A_e(1)B_e(0)F_e(0), \tag{26}$$

where

$$A_e(1)B_e(0) = \left[\begin{array}{cc} 0.7500 & 0.0000 \\ 3.0000 & 0.0000 \\ 0.0000 & 0.0000 \\ -1.0000 & 1.0000 \\ 0.0000 & 0.0000 \\ 1.0000 & 0.0000 \\ \hline & \\ \mathbf{O}_{4\times2} & \end{array}\right] \tag{27}$$

The dimension of a reachable subspace of the pair $(X(0), A_e(1)B_e(0))$ is 5. Since all unreachable eigenvalues of $X(0)$ are zeros, there exists an orthogonal matrix T such that

$$\begin{aligned} TX(0)T^T &= \begin{bmatrix} \mathbf{O}_{5\times5} & \mathbf{O}_{5\times5} \\ X_{21} & X_r \end{bmatrix} \\ TA_e(1)B_e(0) &= \begin{bmatrix} 0 \\ B_r \end{bmatrix} \end{aligned} \tag{28}$$

where the pair (X_r, B_r) is reachable. By the method of [7], we can find a feedback gain $F(0)$ such that all eigenvalues of $X_r - B_r F(0)$ are zeros. Thus, we have a 2-periodic feedback gain as follows.

$$\begin{aligned} F_e(0) &= [\ \mathbf{O}_{2\times5}\ \ F(0)\ ]T \\ &= \begin{bmatrix} -4.9244 & -4.1748 & -0.6243 & 0.6801 \\ -151.7922 & -193.2352 & -20.2396 & 29.0100 \end{bmatrix} \\ &\quad \begin{matrix} -3.9181 & -0.9204 & -0.6320 & 0.6703 & 0 & 0 \\ -105.6955 & -51.2278 & -15.3516 & 24.0320 & 0 & 0 \end{matrix} \end{aligned}$$

$$F_e(1) = 0$$

According to the method of Section 3, we have the 2-periodic Kalman filter with the Kalman gain

$$K(0) = \begin{bmatrix} 0.7020 & 0.5060 \\ 0.1766 & 1.5782 \\ -0.0882 & 0.0846 \\ 0.2978 & -0.5060 \\ 0.3766 & 0.1377 \\ 0.0000 & 0.5000 \\ 0.2768 & -0.0056 \\ 0.0293 & -0.1372 \\ 0.0000 & 0.0000 \\ 0.0000 & 0.0000 \end{bmatrix}$$

$$K(1) = \begin{bmatrix} 0.3114 & 0.2906 \\ 0.0000 & 0.9999 \\ 0.0168 & -0.0001 \\ -0.0272 & -0.2733 \\ 0.3384 & 0.0438 \\ 0.0035 & 0.2942 \\ -0.0592 & 0.0522 \\ 0.1615 & -0.0438 \\ 0.0000 & 0.0000 \\ 0.0000 & 0.0000 \end{bmatrix}$$

For this system, we simulate the state response for the same initial value as [3]. In this result, we get the sum of least square x_1 as 216.8575, it is smaller than the result of [3].

5 Conclusion

In this paper, we have shown the numerical example in the case without noise. Our future study is to simulate the system with system noise and observation noise.

References

[1] W. Zhang, M. S. Branicky and S. M. Phillips, "Stability of networked control systems", *IEEE Control System Magazine*, Vol. 21, No. 1, pp. 84-99, 2001.

[2] D. Hristu, "Stabilization of LTI systems with communication constraints", *Proc. of American Control Conference*, pp. 2342-2346, 2000.

[3] S. Yamamoto, T. Kawagoe and T. Ushio, "Stabilization of control systems with a periodic communication constraint by an observer-based controller"(in Japanese), *Trans. of The Institute of Systems, Control and Information Engineers*, Vol. 15, No. 11, pp. 593-599, 2002.

[4] N. Takahashi, M. Kono and A. Kose, "Stabilization of networked control systems with communication bus under limited bandwidth", *Proc. of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, Vol. 7, pp. 247-252, 2003.

[5] M. Kono, "Eigenvalue assignment in linear periodic discrete-time systems", *Int. J. Control*, Vol. 32, No. 1, pp. 149-158, 1980.

[6] C. E. Souza and G. C. Goodwin, "Periodic solutions of matrix Riccati equation in optimal filtering of nonstabilizable periodic systems", *Proc. of 10th IFAC World Congress*. Vol. 9, pp. 249-254, 1987.

[7] K. Hiranuma and M. Kono, "Design of deadbeat control system on the basis of orthogonal transformations"(in Japanese), *Proc. of the 18th Control Theory Symposium*, pp. 229-232, 1989.

[8] E. W. Kamen and J. K. Su, *Introduction to Optimal Estimation*, Springer, 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Design of Human Support System for Smart Assistance

H. Nishiyama, Y. Mine, W. Yamazaki, M. Obayashi and F. Mizoguchi
Information Media Center
Tokyo University of Science
Noda, Chiba, 278-8510, Japan

Abstract

We designed an agent system for realizing a smart assistance environment in a house or office space. The system supports humans using people's position and action for individualization. We enhanced the ability of sensing in the environment using camera robots and laser sensors for detecting people and their motion. Using such a hardware environment, the laser sensor can detect the exact coordinates of the person and robot, and individual human faces can be recognized using the camera video. Monitoring and support programs are implemented for individuals in order to recognize them and trace their actions.

1 Introduction

Recent developments in networks have enabled us to control several information appliances via the Internet. Similar developments in radio communication have enabled us to employ mobile information machines, such as mobile robots, in office and domestic applications. In turn, these developments have enabled research into the design of smart houses and smart offices for supporting our lives and work. For example, research was conducted for supporting office workers, such as delivery and guidance service using mobile robots[1][3]. However, such research rarely considered information on the location of persons who act within the environment. As a result, the human is treated only as the coordinates, task generator, the target of the task, or an obstacle in executing tasks. Therefore, support is neither tailored to the individual within the environment nor efficient.

To solve these problems, we designed an agent system for realizing a smart assistance environment in a house or office space. The system supports a human using the human's position and action. We enhanced the ability of sensing in the environment using a camera robot and laser sensor for detecting the human and his/her motions. Using such a hardware environment, the laser sensor can detect the exact coordinates of the human and robot, and it is possible to recognize individual faces using the camera. Monitoring and support software for each person is implemented in order to recognize individuals and trace their action. In this paper, we define the software that monitors and supports a person as a support agent, and the software that controls an apparatus as an apparatus agent.

The support agent recognizes the person's action by collecting the person's information from the sensor agent group and judges what support is required. When support is required, the support agent requests an individual apparatus agent to perform an appropriate support task. In a process in which each apparatus agent is performing the support task, the support agents exchange information with those apparatus agents, increasing the efficiency of the support task within the whole environment.

We designed and integrated these agents using the multi-agent language MRL that we developed[4]. MRL introduced the functions of synchronization and asynchronous control included in parallel logical language KL1[2], together with stream communication, and can realize parallel processing of two or more agents, communication between agents, and dynamic generation and deletion of agents. Furthermore, we demonstrated the effectiveness of these agents using the Information Media Center building of Tokyo University of Science as an experiment environment.

2 Approach for Smart Assistance

Smart Assistance provides services for working persons in an office or home environment. The services employ robots and information appliances for smart support of persons' work and for providing information. The person's behavior is traced so the support can be provided automatically by recognizing the type and purpose of the work, and the situation of the person. In this paper, we don't attach any sensors to the people directly to recognize their behavior, but rather integrate information of several sensors installed in the environment. People can thus receive service automatically without feeling mechanically restrained. The assumed support includes providing information on the terminal, controlling the information appliances, and several different services using some type of robot. We also consider personification and complex services. That is, we imagine that the environment of the office or building is one robot and design an action support

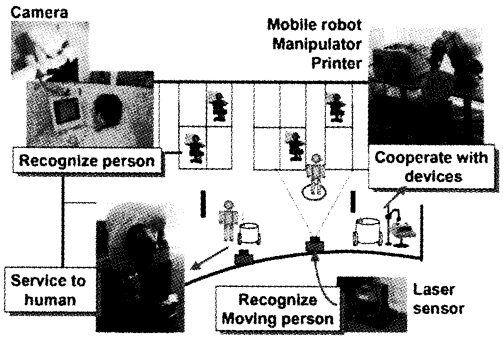


Figure 1: Smart assistance concept

system to suit the person in the environment.

Figure 1 illustrates a practical example of smart assistance. In this figure, mobile robots and manipulators are performing delivery services to support the persons, and laser sensors and camera sensors recognize the persons' locations and situations. By cooperation with these devices and sensors, each robot can perform the service at the person's new location when the person moves after requesting a task.

To realize this smart assistance, we must design a model that has many functions that control several sensors and integrate their information to recognize the environment, people, and a person's behavior. In addition, the model needs a function to integrate control systems of several devices such as information appliances and mobile robots. The problems are as follows.

- Using information from several sensors
  We use laser sensors and camera sensors to trace a person's behavior and recognize a person in the environment. To trace a person's behavior, we must share the information of each different sensor; to achieve real-time recognition, the analysis of each sensor must be performed as soon as possible.
- Parallel and cooperative control of devices such as mobile robots
  Each robot must be controlled in parallel or cooperatively if needed. Therefore, negotiations for selecting a cooperating partner and sharing information for performing cooperative tasks are needed, and real-time scheduling is needed to avoid collisions when the individual is moving.
- Recognizing the situation and supporting the action of people individually
  To recognize the current location and action details individually in the environment, we must collect information from several sensors and assign tasks to devices such as robots and information appliances to support a person's action.

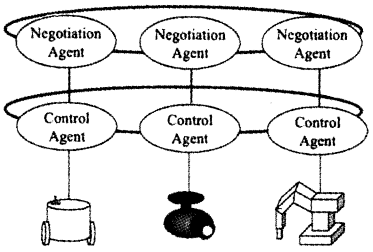


Figure 2: Structure of device agents

To solve these problems, software that controls a device, such as a sensor system, a robot or an information appliance, is defined as a device agent in this paper, and the software that supports a person individually is called a support agent. Here, our approaches are as follows.

- Definition of the two kinds of agents that constitute a device agent
  To achieve efficient information sharing and rapid performance, we divide a device agent into a "control agent" that controls devices directly and a "negotiation agent" that executes intellectual processes such as negotiation and scheduling as shown in Fig. 2. The computer connected to each apparatus does not need to be highly efficient, and processing can be distributed by using a computer group such as a Grid computer.
- Dynamic generation of the support agent
  To recognize an individual person, we define an agent that monitors the behavior and supports the actions of each recognized person. We call this agent the support agent. This agent is generated when a new person is recognized in the environment based on information of several sensors; it is deleted when the person moves out of the environment. The generated support agent receives information from the device agents of the sensors and continues tracing the person's behavior. The support agent requests device agents to perform services for the person based on the tracing results.

As mentioned above, we not only define several kinds of devices and sensors as an individual agent, but also define the support agent that monitors and supports a person. A person who has been recognized within the environment will thus be monitored and supported by the support agent, and the person can receive individual services based on the judgment of the support agent.

3 System Design

In order to realize the above-mentioned smart assistance, it is necessary to model the distribution and cooperation of two or more agents. To implement these agents, we applied a multi-agent language MRL (Multi-agent Robot Language) [4] based on a parallel-logic programming language. The MRL supports agent communication through broadcasting by extending synchronous and asynchronous control. It also enables stream communication, allowing parallel execution of multi-agents and dynamic generation and deletion of agents. Using the MRL, we can define each device and sensor as an individual agent, and these agents can be generated or deleted as necessary.

3.1 Device Agent

We define the following two kinds of agents to control devices.

3.1.1 Control Agent

A control agent sends commands to devices and receives information such as sensing data from devices through "Control software." A control agent is able to open a direct communication channel with another control agent and can exchange messages with it while cooperating.

3.1.2 Negotiation Agent

A negotiation agent receives task requests from another negotiation agent or user in the environment. When a control agent requests a cooperative task, the negotiation agent negotiates with another negotiation agent to select a cooperating partner. As a result of negotiation, a negotiation agent that is committed to a task sends a description of the task to its own control agent.

3.2 Support Agent Generator

A support agent generator broadcasts sensing data from sensor device agents to support agents as shown in Fig. 3. The support agents then reply with candidate data of the person it is monitoring in the received data group. When the support agent generator does not receive a response, it judges that the data is that of a new person in the environment and generates a support agent to support that person. In addition, when received data is redundant from some support agents, the support agent generator notifies these support agents. Thus, the support agent generator not only generates the support agents, it also is a supervisor that recognizes collisions between support agents.

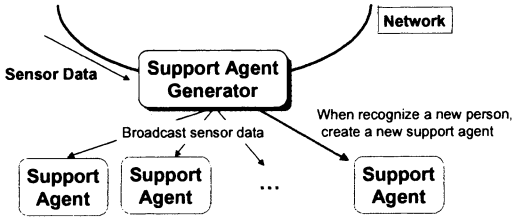


Figure 3: Support Agent Generator

3.3 Support Agent

A support agent monitors and supports a person, and collects information needed for smart assistance. The agent makes requests to device agents based on this information and performs services for the person. In this paper, we define the following personal information that a support agent has.

$$H = \{ID, Location, Behavior, Service\}$$

Each variable is defined below.

- *ID* ... Identification of a person
  When a person is not recognized (initial state), *unknown* is substituted.
- *Locate* ... Location data of the person
  This information is used by an execution service and also affects the priority of services based on the distance relation with robots.
- *Behavior* ... Person's behavior data
  Examples of data include whether the person is in a room (working), when the person is moving in the environment, and the direction in which the person is moving.
- *Service* ... The history of services performed
  This information affects the priority of services.

Information without *Service* is updated dynamically by the sensing data, and device agents are requested to provide several services based on the information.

4 Implementation and experiment

Figure 4 depicts the relation with each agent in our implemented system. The device agents consist of robot and sensor system agents, and the support agents integrate these device agents.

4.1 Sensor System Agents

The sensor system agents that we implemented are as follows.

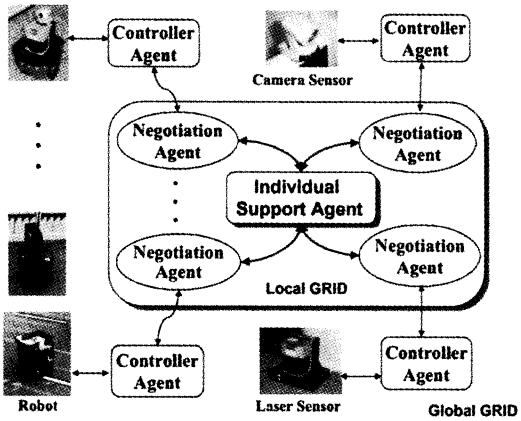


Figure 4: Connection between each agent

- Laser sensor agent
  This agent controls wide range laser sensor, and repeats laser irradiation. It recognizes the coordinates of the dynamic obstacle of irradiation within the limits and discriminates the kind of obstacle from the reflected irradiation pattern.
- Camera sensor agent
  This agent sends a view from the camera when it receives a request message from other agents. Usually, this agent is used for a fixed camera on the terminal used by the person, and the view is used for individual recognition of the person and for TV meetings between other terminals.
- Face identification agent
  This agent performs individual identification by video analysis when it receives a request from a camera sensor agent. Identification is performed through template matching with previously registered image data. The template uses a correlation value and the distance between coordinates to specify the individual using eye, nose, and mouth features.
- Keyboard and mouse sensor agent
  If the keyboard or mouse is being operated when this agent receives a request from another agent, this agent reports the fact to the requesting agent. This function is usually used as a switch for the above identification by a face view.

4.2 Individual support via the support agent

A support agent is generated when a person is recognized in the environment. This support agent exists as long as the person stays in the environment and is deleted when the person exits the environment. The

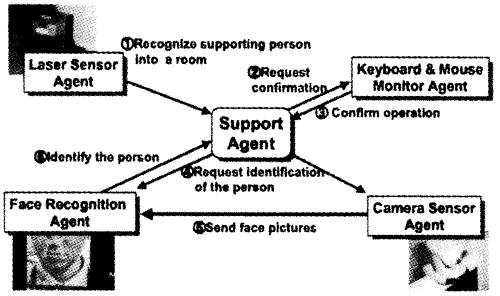


Figure 5: Flow of messages in identifying a person

person's data is saved in a database when the agent is deleted, and the data is loaded when the person is re-recognized. This agent performs communication for identifying the recognized person (refer to Fig. 5) and for negotiating with other agents when the supporting person requests any services. In addition, when the person moves in the environment while services are being performed, this agent negotiates with each other agent so that the service may be modified according to the person's behavior. Consequently, the movement is traced based on the information acquired from the laser sensor agent, and printed paper is delivered to the new location as illustrated in Fig. 1, even when the person moves after making a service request.

5 Summary

In this paper, we designed an agent system that monitors people's position and behavior information within the environment and supports them by providing smart assistance. By using this system, the people can receive automated service from the environment without feeling any mechanical restraint.

References

[1] W. Burgard, A. B. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner and S. Thrun, Experiences with an interactive museum tour-guide robot, *Artificial Intelligence*, Vol. 114, pp. 3-55, 1999.

[2] T. Chikayama: "A KL1 Implementation for Unix Systems", *New Generation Computing*, Vol.12, pp.123-124, 1993.

[3] F. Mizoguchi, H. Nishiyama, H. Ohwada and H. Hiraishi, Smart Office Robot Collaboration based on Multi-agent Programming, *Artificial Intelligence*, Vol. 114, pp. 57-94, 1999.

[4] H. Nishiyama, H. Ohwada and F. Mizoguchi, A Multiagent Robot Language for Communication and Concurrency Control, *International Conference on Multiagent Systems (ICMAS98)*, pp.206-213, 1998.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Dynamically Separating Learning Algorithm for Interactive Computers in Dynamic Environment : Optimization of collective performance in networked computers

Koichi NAKAYAMA[1),2)] Katsunori SHIMOHARA[1),2)] Osamu KATAI[1)]
[1)]Graduate School of Informatics
Kyoto Univ. Kyoto, 606-8501, Japan
[2)]Human Information Science Labs.
ATR. Kyoto, 619-0288, Japan

Abstract

We have previously proposed the dynamically separating learning algorithm (DS-LA), which separates networked computers into some sub-networks and dynamically changes each sub-network's structure. In this paper, we verify experimentally that, when applying the DS-LA to computers in a dynamically changing network environment, the DS-LA optimizes each computer's parameters for the network so that the collective performance of the networked computers becomes the maximum. The experimental results suggest that DS-LA is effective in a ubiquitous computing environment, where the network environment changes unpredictably.

1 Introduction

In recent years, various types of interactive computer networks have been researched. In this environment, networked computers affect each other's performance mutually. The optimal state for one computer's performance, however, may not be the optimal state for another's or for the collective performance for all the networked computers. This is called a dilemma environment [1] for computers. Since each computer cannot perceive all of the influences across the whole network, it is difficult to optimize the networked computers in the dilemma environment. Particularly, an environment called ubiquitous computing [2] attracts attention. Ubiquitous computing means an environment in which many interactive computers exist, and computers operate both cooperatively and autonomously. In this environment, the number of the networked computers, the type of computers, the method of communication, and etc. are changing dynamically. In other words, ubiquitous computing environment is both a dilemma environment and a dynamic environment. It is difficult to predict all the optimal values for the parameters and them set up beforehand. However, when optimizing the collective performance in the whole network, it is important to optimize the network parameters that affect other computers throughout the network. Thus, we require an optimization algorithm for collective performance, e.g., learning algorithm, that optimizes the network parameter of each computer.

We have proposed dynamically separating learning algorithm (DS-LA) [3][4] to optimize the networked computers. When DS-LA is applied to statically networked computers, the characteristics of DS-LA are verified with conventional research [5][6]. The DS-LA has ability to set the computers' parameters for optimal collective performance under a static dilemma environment.

In this paper, we verify experimentally that the DS-LA keeps optimizing the network parameters so that the collective performance of all the networked computers is at its maximum under not a static, but a dynamic, dilemma environment. As a result, we show that DS-LA is valid for ubiquitous computer networks.

2 Dynamically Separating Learning Algorithm (DS-LA)

The DS-LA separates the computers' interaction dynamically into sub-networks called channels, shown in Fig. 1. A channel is an imaginary network in which interactions are possible. DS-LA is realized by DS-LA programs in the networked computers. Each hardware has not the same characteristics. The hardware of computers is unchangable by learning.

We provide each computer with a DS-LA program. The total number of computers is N_A. Each DS-LA program contains the following parts: The DS-LA program in computer a has network parameters $Pn(a)$, an affiliation channel number $Ch(a)$, and a self-evaluation value $E(a,t)$. Network parameters $Pn(a)$ are current parameters that affect other computers throughout the network. The affiliation channel number $Ch(a)$ is a current number that represents which channel com-

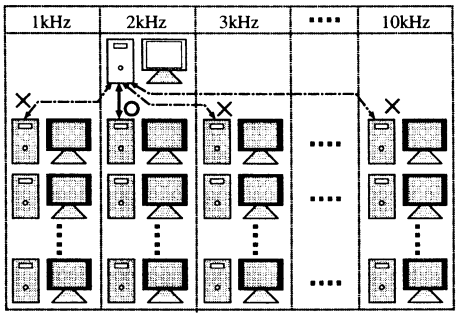


Figure 1: Schematic diagram of separated computers. Each computer interacts only with a computer in the same channel.

puter a belongs to. A computer cannot interact with another computer with a different channel number. A self-evaluation value $E(a,t)$ in computer a at time t is a value that depends on each computer's performance $Pf(a,t)$. The evaluation value changes according to Eq. 1,

$$E(a,t+1) = (1-\alpha)E(a,t) + \alpha Pf(a,t), \qquad (1)$$

where α is a learning coefficient.

Each computer interacts according to the models that are shown in Section 3 and Section 4 for every unit time. The computer's performance is determined by the interaction in the model. Moreover, the DS-LA program of each computer carries out the following operations for every unit time.

(1) A DS-LA program choses, by P_{Up} of probability, the optimal computer r_{best} (the one with the best self-evaluation value) out of $N_{Candidate}$ computers with the same characteristics, which are chosen randomly in all the channels. If $E(a,t) > E(R_{best},t)$, DS-LA program copies the parameters' values of computer r_{best} to the parameters' values of computer a as in Eq. 2. At this time, the parameters' values are mutated by P_{mut} of probability

$$\begin{aligned} Pn(a,t+1) &= Pn(r_{best},t) \\ Ch(a,t+1) &= Ch(r_{best},t). \end{aligned} \qquad (2)$$

(2) When the number of computers in the same channel is more than N_{lim}, half of these computers are assigned to a new channel. If there is only one computer in a given channel, the computer's DS-LA program copies network parameters and channel number of other computers to itself, similarly to (1).

In DS-LA, the network structure is decided by each computer's channel number. The channel number can be considered one type of network parameter that is restrained by DS-LA. Because of that, we verify network parameters and channel number independently. As a result, each DS-LA program optimizes the network parameters and the channel number, so that the collective performance of all the computers $\sum_{a=0}^{N_a} Pf(a,t)$ is maximized, where the number of combination of the network parameters and the channel number is $(Pn_{sum})^{N_a}$ and $(Ch_{sum})^{N_a}$, respectively. The details of DS-LA are shown in [4][6].

We set the parameters as follows. $N_A = 100,000$, $N_{Lim} = 10$, $P_{mut} = 0.01$, $P_{Up} = 0.2$, $N_{Candidate} = 10$, $\alpha = 0.9$, $\forall a, E(a,0) = 0$. The initial network parameter $Pn(a)$ and the initial channel number $Ch(a)$ are chosen at random. Furthermore, 100 runs were performed with different random seeds.

3 Optimization of the network parameters

In this section, we verify experimentally under a dynamic environment that DS-LA can optimize the network parameters.

A network parameter affects other computers throughout a network. Examples of parameters include packet size, the amount of communications, and the amount of task requests, etc. In some cases, these parameters cause a dilemma environment. Here, a packet size[7] of the dynamic dilemma environment is explained as an example.

In packet transmission, if a computer comunicates with large packets, once it is connected, many packets can be sent at once. If all the computers communicate using large packets, each computer must wait a long time, thus collective performance decreases. This is in a dilemma environment. This section describes the acquisition of an optimal network parameter for the collective performance according to the network states and the characteristics of each computer's hardware in a dynamic environment.

3.1 Experimental Model

To verify that DS-LA optimizes network parameters under a dynamic dilemma environment, we define each computer's performance at unit time t by Eq. 3,

$$\begin{aligned} Pf(a,t) &= Pn(a,t) \\ Pf(r,t) &= -2(Pn(a,t) - S_{Char(a)}) \\ &\quad (if Pn(a,t) > S_{Char(a)}), \end{aligned} \qquad (3)$$

where r is a randomly chosen computer from the same channel. $Char(a)$ is the hardware characteristic of computer a, that are not all the same. S_{Char} is a threshold defined by the hardware's characteristic that decides the influence on the other computers. In a dynamic environment, threshold S_{Char} for the hardware characteristic dynamically changes as follows. In 0 - 200 unit times, $\{S_A$=2, S_B=4, S_C=6, S_D=8$\}$ and in 200 - 400 unit times, $\{S_A$=8, S_B=6, S_C=4, S_D=2$\}$.

In this dynamic dilemma model, computer a obtains high performance when a network parameter $Pn(a,t)$ is high. However, when the network parameter $Pn(a,t)$ exceeds the threshold $S_{Char(a)}$, the performance of the computer r decreases. Namely, the optimal value of the network parameter $Pn(a,t)$ for high collective performance is equal to the threshold $S_{Char(a)}$. Details of the experimental model are shown in [6][4]. We set the parameters as follows. $\forall a$, $Char(a) \in \{A, B, C, D\}$, $Ch(a) \in \{1, 2, 3 \cdots N_{Ch_{Sum}}\}$, and $Pn(a) \in \{1, 2, 3 \cdots 10\}$.

3.2 Experiments and Discussion

In all 100 runs, we gained the following results. The history of the population ratio of each computer's network parameter is shown in Fig. 2, and the history of the average performance of the whole computer network with respect to collective performance is shown in Fig. 3.

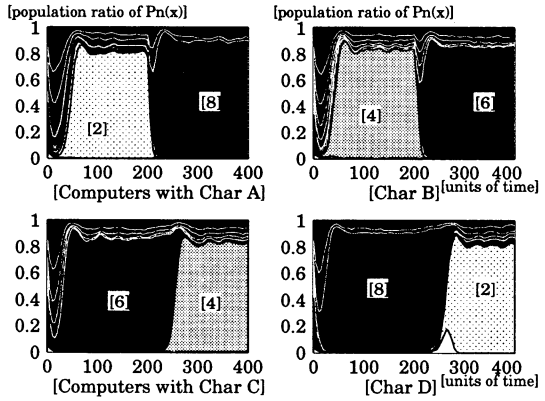


Figure 2: History of the population ratios of network parameters. Four figures show the four types of hardware characteristic's history for the population ratios. The ratio of the network parameter is represented by the distance between lines. After 50 units of time, the optimal parameters for the collective performance occupied most of the population in every characteristic. At 200 units of time, the optimal parameters for the collective performance change, but after the 250 units of time, they occupied most of the population again.

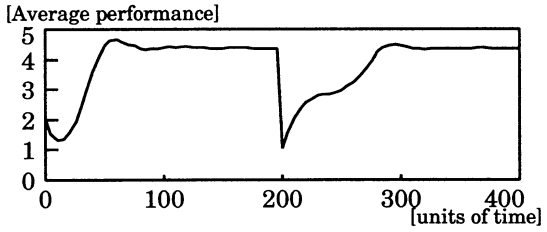


Figure 3: History of the average performance of a whole computer network with respect to collective performance. In this model, the optimal collective performance is five. At about 50 units of time, the average performance reaches about 90 % of the optimal collective performance. At 200 units of time, the optimal network parameters for collective performance changes, and the average performance falls. However, after 250 units of time, the average performance recovers to more than 90 % of optimal collective performance.

The details of the mechanism to acquire the optimal parameters for collective performance until 200 units of time by dynamic separation is shown in [6][3]. When computer x with $Char(x) \in \{A, B\}$ acquires the optimal parameter for the collective performance after the 200 units of time, the mechanism explained in Fig. 4 occurs.

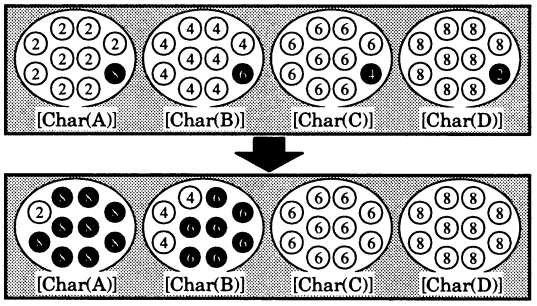


Figure 4: Schematic explanation of the mechanism by which computer x with $Char(x) \in \{A, B\}$ acquires the optimal parameter for collective performance. For computer x with Char A, since S_A changes from 2 to 8 at 200 units of time, computer x with $Pn(x) = 8$ has higher optimality, not only for a computer's performance, but also for the collective performance, than computer x' with $Pn(x') = 2$. Accordingly, when network parameter $Pn(x)$ changes to 8 by a mutation, computer x obtains a high chosen probability by updating the computers. Thus, the ratio of computer x with $Pn(x) = 8$ increases. The explanation is similar for computer x with Char B.

On the other hand, when computer x with $Char(x) \in \{C, D\}$ acquires the optimal parameter for the collective performance after the 200 units of time, the mechanism explained in Fig. 5 occurs.

Using DS-LA, the computer learned the parameter

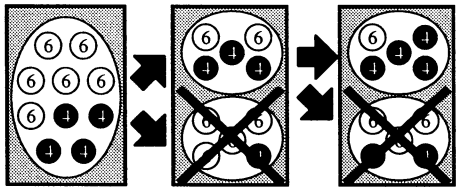


Figure 5: Schematic explanation of how the mechanism that computer x with $Char(x) \in \{C, D\}$ acquires the optimal parameter for collective performance. For computer x with Char C, since S_C changes from 6 to 4 at 200 units of time, when two or more paramters $Pn(x)$ change to 4 by a mutation in a channel, the channel has a higher collective performance than another channel that is occupied by $Pn(x') = 6$. When the collective performance in a channel is higher, the number of computers in the channel increases faster and the channel is dynamicaly separated earlier. Because more computers with higher self-evaluation exit in such a channel, the computers in the channel are chosen with higher probability. With the dynamic separation, one of the separated channels has a higher ratio of computer $Pn(x) = 4$ than the others by a certain probability. As a result of the above effects, computers acquire parameters that achieve a high collective performance.

that achieved the highest collective performance in the dynamic environment.

4 Optimization of the network structure

In this section, we verify experimentally under a dynamic environment that DS-LA can optimize the channel number as a network parameter which is restrained by DS-LA.

A set of the computer's channel numbers show the structure of the network.

In some cases, the channel number causes a dilemma environment, similar to the other network parameters. For example, the network has three types of computers, which perform a task cooperatively; one has only image DSP boards, another has only sound DSP boards, and the other has only Fast Fourier Transform (FFT) DSP boards.

When an image processor and a FFT processor are simultaneously required for processing a task, it is efficient when a computer with an image DSP board and a computer with a FFT DSP board cooperate in the same channel. However, when the same types of computers exist in the same channel, since they can be requested their task to others, the calculation-cost decreases. Moreover, when two or more sound processors are simultaneously required to process a task, it is efficient when only computers that have a sound DSP board cooperate within the same channel.

This section describes the acquisition of an optimal combination of computers for collective performance in channels according to the characteristics of each hardware in a dynamic environment.

4.1 Experimental Model

To verify that DS-LA optimizes the channel numbers for network structure under a dynamic dilemma environment, each computer's performance at unit time t is defined in Table 1.

Table 1: Computers' performance table by interaction.

Pf(a) \ Pf(r)	A	B	C
A	2 / 0	1 / 3	2 / 0
B	1 / 3	2 / 0	2 / 0
C	2 / 0	2 / 0	1 / 3

Before 200 units of time

Pf(a) \ Pf(r)	A	B	C
A	1 / 3	2 / 0	2 / 0
B	2 / 0	2 / 0	1 / 3
C	2 / 0	1 / 3	2 / 0

After 200 units of time

For every unit time, computer a interacts with computer r chosen rondomly from the same channel. According to the characteristics of computers a and r, Table 1 decides the performance of computer a and computer r, where a computer has one of three types of hardware characteristics: $\forall a, Char(a) \in \{A, B, C\}$, and each computer has a channel number, $\forall a, Ch(a) \in \{1, 2, 3 \cdots Ch_{Sum}\}$; however, the network parameter $Pn(a)$ is omitted. Before 200 units of time, when computers with Char A and those with Char B exist in the same channel, and when computers with the same characteristics of C exit in the same channel, the collective performance is optimal for all computers, although each computer's individual performance is not optimal. After 200 units of time, the interaction's performance of A and C interchanges.

Details of the experimental model are shown in [6][4].

4.2 Experiments and Discussion

In all 100 runs, the results were as follows. The history of the average population ratio for each characteristic in the same channel is shown in Fig. 6.

The details of the mechanism to acquire the optimal combination of computers for collective performance until the 200 units of time by dynamic separation is shown in [6][3]. The mechanism to acquire the optimal combination of computers for collective performance after the 200 units of time can be explained in Fig. 7.

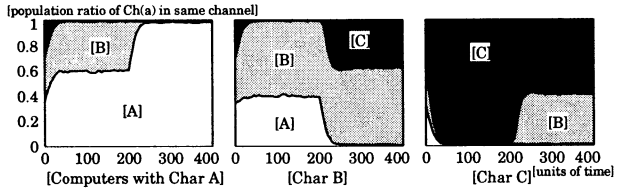


Figure 6: History of the average population ratio of each characteristic in the same channel. The three figures show the history for the three types of hardware characteristics. Before 200 units of time, two-third of the channels are occupied by computers a with $Char(a) \in \{A, B\}$, while one-third of them are occupied by $Char(a) = C$. After 200 units of time, one-third of the channels are occupied by $Char(a) = A$, while two-third of them are occupied by $Char(a) \in \{B, C\}$. Most of the channels acquired the optimal combination of computers for optimal collective performance.

By using DS-LA, the computers could learn the optimal combination of computers for collective performance in a dynamic environment.

5 Conclusion

In this paper, examined a network parameter and network structure as factors that affect collective performance in networked computers. Using a model that has taken into account factors from dilemma and dynamic environments, we verified the validity of DS-LA for networked computers. Results show that even when the optimal network parameter and network structure for collective performance changes dynamically, the computers that applied DS-LA learned the optimal network parameter and network structure. These results suggest that DS-LA is effective in a ubiquitous computing environment where the network environment changes unpredictably.

Acknowledgements

This research was supported in part by the Telecommunications Advancement Organization of Japan.

References

[1] Robert Axelrod, "The Evolution of Cooperation," *Basic Books*, 1984.

[2] Mark Weiser, "The Computer for the 21st Century," *Scientific American*, Vol. 265, No. 3, pp. 99-104, 1991.

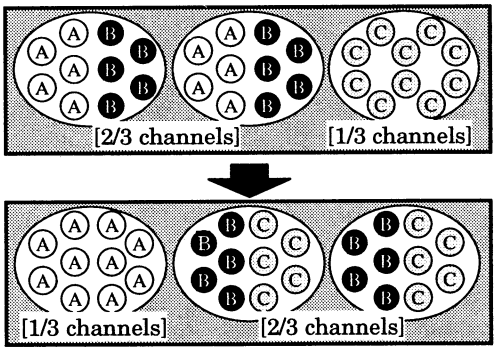


Figure 7: Combination of Computers in Channels. After 200 unit time, in the channels that computers a with $Char(a) \in \{A, B\}$ belong to, the rate of increase of channels that have more population ratio of $Char(a) = A$ is higher. About channels that have $Char(a) = C$, the rate of increase of channels that have more population ratio of $Char(a) = B$ is higher. As a result of the above effects in combination, $Char(a) = B$ move to the channel that has $Char(a) = C$ from the channel that has $Char(a) \in \{A, B\}$.

[3] K. Nakayama, H. Matsui, Y. Nomura, "Dynamically Separating GA (DS-GA)," *IPSJ Transactions on Mathematical Modeling and Its Applications (In Japanese)* , Vol. 43, No. SIG 10 (TOM 7), pp. 95-109, 2002.

[4] K. Nakayama, H. Matsui, K. Shimohara, O. Katai, "Dynamically Separating Learning Algorithm for Interactive Computers (DS-LA$_{IC}$)," *Proceedings of the Sixth International Conference on Human and Computer* , pp. 182-187, 2003.

[5] K. Nakayama, K. Shimohara, O. Katai, "Dynamically Separating GA: A New Method of Achieving the System-level Optimality in MAS," *The journal of three dimensional images*, Vol.16, No.4, pp. 177-183, 2002.

[6] K. Nakayama, H. Matsui, K. Shimohara, O. Katai, "Proposal of Dynamically Separating Learning Algorithm for Interactive Computers (DS-LA): Optimization of collective performance in networked computers," *The journal of three dimensional images*, at press.

[7] Y. Matsumoto, Y. Takahashi and T. Hasegawa, "The Effects of Packet Size Distributions on Output and Delay Processes of CSMA/CD," *IEEE Transactions on Communications* , Vol.38, No.2, pp. 199-214, 1990.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

On a model of motion sickness induced by Coriolis stimuli

Naoki Kitayama and Mieko Tanaka-Yamawaki
Department of Information and Knowledge Engineering, Tottori University
Tottori, 680-8552, Japan

1. Introduction

Motion sickness is familiar to us in various situations such as traveling by boats, cars, or space-crafts, or watching movies. Coriolis stimulus is often used to artificially create the situation of such motion sickness effectively in the laboratory. Coriolis stimulus can be applied to a subject by simultaneously giving an pendulum-like oscillation on its head with a horizontal rotational motion. Under such situation the subject receives an illusion as if he/she moves toward a certain direction which is untrue. Isu and his collaborators have named this sensation as Corioli illusion. [5-7] They set the experimental situation in order to minimize the Coriolis acceleration by fixing the center of a head right on the horizontal rotational axis and move the head back and forth around the axis connecting the ears. By doing this, inner ears are kept almost still and centrifugal acceleration is kept very small.

Discussions related to the response of semicircular canals and the sense of rotation are found in the literatures [1--7]. In the present paper, we examine a model of semicircular canals proposed in some of the previous literatures [1,6] at large in order to draw a systematic explanation of a result which one of the authors (N. K.) obtained in the course of participating the experiments designed by Isu et. al. We particularly focus on the quantitative relationship between the applied stimulus and the response from the subjects on the sickness caused by the Coriolis stimulus.

2. Modeling semicircular canal system

We describe the motion of endolymph inside each semicircular canal and estimate the degree of sensation of rotation by each semicircular canal when the head is rotated. We first notice that the rotational axis of the unified senses mismatches with the rotational axis of the applied stimulus [6,7]. This is due to the fact that the three semicircular canals (anterior, posterior, and horizontal) are not exactly perpendicular each other. [8,9].

In the process of unifying the sensations in the central nervous system, we suppose that the connection of nerves are balanced in such a way that the axis of rotation applied by the stimulus force overlaps with the axis of rotation that our sensation receives.[6,7] In doing this we can derive the correct result independent of the anatomical position of the semicircular canals even if the semicircular canals are not exactly perpendicular.

Thus we consider a model in which the three semicircular canals (anterior, posterior, and horizontal) are exactly situated on the frontal plane, on the sagittal plane, and on the transverse plane, respectively. We also incorporate the experimentally measured values of time constants for sensation of rotation (of a head around x-axis, y-axis, z-axis) [10,11] and also the time constant of vestibular oculomotor reflex[11-13].

3. Rotation of semicircular canals

Coriolis stimulus is applied to a subject by rotating the subject on a horizontally rotating table, and oscillating the head around the axis connecting both ears. We set the time to start moving the head is t=0, and the rotation angle of the horizontal table at time is $\psi(t)$, and the angle of rotating the head is $\theta(t)$.

Also the angular velocity can be expressed as $\dot{\psi}(t)$ and $\dot{\theta}(t)$, respectively.

Let $O_0(x,y,z)$ represents the coordinate system fixed to the laboratory, $O_1(x',Y,z)$ be the coordinate system fixed to the horizontally rotating table, and $O_2(X,Y,Z)$ be the system fixed to the head. O_1 is obtained by rotating O_0 around z-axis by angle $\psi(t)$, represented by a matrix R_1, and O_2 is obtained by rotating O_1 around Y-axis by angle $\theta(t)$, represented by a matrix R_2. They can be written as

$$R_1 = \begin{bmatrix} \cos\psi(t) & \sin\psi(t) & 0 \\ -\sin\psi(t) & \cos\psi(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$R_2 = \begin{bmatrix} \cos\theta(t) & 0 & -\sin\theta(t) \\ 0 & 1 & 0 \\ \sin\theta(t) & 0 & \cos\theta(t) \end{bmatrix} \quad (2)$$

$$\omega_2(t) = R_2\left(\begin{bmatrix} 0 \\ \dot{\theta}(t) \\ 0 \end{bmatrix} + R_1\begin{bmatrix} 0 \\ 0 \\ \dot{\psi}(t) \end{bmatrix}\right) = \begin{bmatrix} -\dot{\psi}(t)\sin\theta(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t)\cos\theta(t) \end{bmatrix} \quad (3)$$

In Eq.(3) the subscript 2 of $\omega_2(t)$ stands for the 2 of the coordinate system O_2., etc.

In order to describe the endolymph motion, we define the coordinate system O_X, O_Y, O_Z (X, Y, or Z, corresponds to the frontal plane, the sagittal plane and the transverse plane of semicircular canals, respectively) fixed to the each semicircular canal. Since the canals are fixed to the head, O_X, O_Y, O_Z maintain the same relative position to O_2. Also the angular velocity of each semicircular canals coordinate system O_* can be expressed by the same expression as Eq. (3).

4. Relation between the motion of endolymph and sensation of rotation

In order to describe the motion of endolymph inside the semicircular canals, we regard endolymph as a rigid body and treat a canal as a uniform semicircular tube. We also assume that the three axes ξ, η, ζ coincide with the principal inertial axes of endolymph and denote the three moments of inertia by I_ξ, I_η, and I_ζ. We further assume that endolymph rotates only around the ζ-axis, which derives

$$I\xi = I\eta \quad (4)$$

and the moment of inertia around ζ-axis moves along the motion of endolymph inside the semicircular canal by means of the viscosity of endolymph and elasticity of cupula.[14]

$$M\zeta = -B\dot{\varphi}(t) - K\varphi(t) \quad (5)$$

Let the rotation angle and angular velocity of relative movement of endolymph at coordinate system fixed to semicircular canals be $\varphi(t)$ and $\dot{\varphi}(t)$, respectively. Denote the angular velocity of rotation in Eq.(3) by $\omega = [\omega\xi\ \omega\eta\ \omega\zeta]^T$, and the moment acting on endolymph by $M = [M\xi\ M\eta\ M\zeta]^T$, rotation of coordinate system fixed to the semicircular canals and moment in endolymph by ω and M, respectively. Then the angular momentum of endolymph can be written as

$$H = \left(I\xi\omega\xi, I\eta\omega\eta, I\zeta(\omega\zeta + \dot{\varphi}(t))\right)^T \quad (6)$$

and we can write down the equation of motion as follows.

$$\frac{d'}{dt}H + \omega \times H = M \quad (7)$$

Since we assume that endolymph rotates only around the ζ-axis, equation of motion around the ζ-axis derived from Eq. (7) is reduced to

$$I\zeta\frac{d}{dt}(\omega\zeta + \dot{\varphi}(t)) - (I\xi - I\eta)\omega\xi\omega\eta = M\zeta \quad (8)$$

Substituting Eq.(4) and (5), we finally obtain the following second order differential equation describing the rotational motion of the endolymph system.

$$\frac{d^2\varphi(t)}{dt^2} + \frac{B}{I\zeta}\frac{d\varphi(t)}{dt} + \frac{K}{I\zeta}\varphi(t) = -\frac{d\omega\zeta}{dt} \quad (9)$$

Setting the initial condition at t=0 to be

$\varphi(0)=\varphi_0$ and $\dot{\varphi}(0)=\dot{\varphi}_0$, we solve Eq.(9) to obtain the angle of endolymph at time t as follows.

$$\varphi(t)=\frac{\tau_1\tau_2}{\tau_2-\tau_1}\left\{\left(-\int_0^t\frac{d\omega_\zeta}{du}e^{\frac{u}{\tau_2}}du+\frac{\varphi_0}{\tau_1}+\dot{\varphi}_0\right)e^{-\frac{t}{\tau_2}}\right.$$
$$\left.-\left(-\int_0^t\frac{d\omega_\zeta}{du}e^{\frac{u}{\tau_1}}du+\frac{\varphi_0}{\tau_2}+\dot{\varphi}_0\right)e^{-\frac{t}{\tau_1}}\right\} \quad (10)$$

Here,$\tau_1+\tau_2=B/K$ and $\tau_1\tau_2=I_\zeta/K$ and τ_1 and τ_2 are the time constants of rotational motion of endolymph.

Following the standard way we apply the subject uniform rotational motion on the horizontal table for some time (30 seconds - 1 minute or more) before oscillating the head. This helps the subject to lose the sense of horizontal rotation at the time of starting the oscillation of the head and thus endolymph stays still relative to the canal, allows us to set the initial values of rotation angle and the angular velocities to be zero, $\varphi_0=0$, $\dot{\varphi}_0=0$. By using the fact $\tau_1 \ll \tau_2$[14], we can write the Eq. (10) by neglecting the second term as follows

$$\varphi(t)=-\tau_1\int_0^t\frac{d\omega_\zeta}{du}e^{\frac{u-t}{\tau_2}}du \quad (11)$$

If we assume that sensation of rotation $\tilde{\omega}_*(t)$ generated by semicircular canals is proportional to the rotational angle $\varphi(t)$ of endolymph and set the constant of proportion be $1/\tau_1$, we obtain

$$\tilde{\omega}(t)=-\int_0^t\frac{d\omega_\zeta}{du}e^{\frac{u-t}{\tau_2}}du \quad (12)$$

where we have suppressed the subscript * to ω denoting a coordinate system X,Y,Z.

Now we will express the total sensation of rotation as a result of integrating the sensations of all the semicircular canals system.

In the coordinate system O_2 fixed to the head.

$$\tilde{\omega}_2(t)=(\tilde{\omega}_X(t),\tilde{\omega}_Y(t),\tilde{\omega}_Z(t))^T \quad (13)$$

where

$$\tilde{\omega}_X(t)=-\int_0^t\frac{d\omega_X}{du}e^{\frac{u-t}{\tau_X}}du$$

and the corresponding relations obtained by replacing X by Y, or Z. Here τ_X, τ_Y, and τ_Z represents the time constant τ_2 around the X-axis, Y-axis, and Z-axis, respectively.

5. Sensation of rotation in an experiment

We discuss the result of an experiment done by applying Coriolis stimulus to the head by means of reciprocating motions. After eliminating the sense of rotation by horizontally rotating the table with a constant angular velocity $\dot{\psi}_0$, we oscillate the head along a sine curve with frequency f starting from the angle θ_0 to the backward direction by angle θ_1 and after a short pose for T seconds, we move the head toward the forward direction by angle θ_2 then stop.

From Eq.(3), the angular velocity of sensation of rotation of a semicircular canal can be written as

$$\omega_2(t)=\begin{bmatrix}-\dot{\psi}_0\sin\theta(t)\\ \dot{\theta}(t)\\ \dot{\psi}_0\cos\theta(t)\end{bmatrix} \quad (14)$$

Thus, by using Eq. (11), the sensation of rotation coming from the semicircular canal system can be estimated to be

$$\tilde{\omega}_2(t)=\begin{bmatrix}-\dot{\psi}_0\int_0^t\dot{\theta}(u)\cos\theta(u)e^{\frac{u-t}{\tau_X}}du\\ -\int_0^t\ddot{\theta}(u)e^{\frac{u-t}{\tau_Y}}du\\ -\dot{\psi}_0\int_0^t\dot{\theta}(u)\sin\theta(u)e^{\frac{u-t}{\tau_Z}}du\end{bmatrix} \quad (15)$$

5. Experiment

We used nine subjects (three men and six women) who has no disorder in their inner ears. We used two values for θ_1, for each value of which we applied 13 different sets of other parameters of Coriolis stimulus (Table 1,2). Parameters are adjusted in such a way that the accumulated angular velocity takes its

minimum value at the end of Coriolis stimulus.

In order to analyze our experimental result by means of absolute evaluation method, we have applied the subjects the standard stimulus for six times before the experiment. We set the average value of unpleasant feeling caused by the standard strength of stimulus to be the standard strength '10'. Also we have asked the subject to give us the evaluated value by any non-negative real number with no upper limit. We have shielded the subjects from vision and sound during the experiment in order to eliminate information coming from outside.

Table 1 Coriolis stimulus with θ_1=15[deg]

θ_2	12.1	10.9	8.9	6.7	4.4	2.2
T	0	0.6	1.8	3.4	5.8	9.8
$\tilde{\varphi}$	29	40	60	80	100	120

Table 2 Coriolis stimulus with θ_1=20[deg]

θ_2	16.2	14.2	12.1	9.8	7.7	5.5	3.2
T	0	0.8	1.7	2.9	4.3	6.2	9.2
$\tilde{\varphi}$	39	60	79	101	120	140	160

6. Result

We conducted our experiments as follows. We have collected verbal responses from each subject on the degree of unpleasant feeling under the Coriolis stimulus at several degrees of strength and computed mean and standard deviation. Then by using them we have normalized each subject's response and average them over all the subjects, for each degree of strength of stimulus. The result is shown in Figure 1.

Therefore we conclude that Coriolis illusion is caused by a mismatch of information coming from different sensory organs such as otolithic system, semicircular canal system, and somato-sensory system. In other words, we insist that the Coriolis illusion can be regarded as an important factor of motion sickness. Moreover, we propose a view that the degree of Coriolis illusion directly reflects the effect of sensory contradiction caused by the motion sickness.

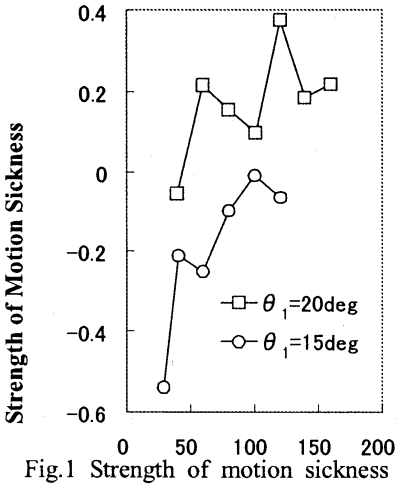


Fig.1 Strength of motion sickness induced by Coriolis illusion is plotted as a function of accumulated angle of rotation $\tilde{\varphi}$ [deg].

References

[1] Groen JJ: The problems of the spinnig top applied to the semicircular canals. Confin Neurol 21: 454-455,1961

[2] Peters RA: Dynamics of the vestibular system and their relation to motion perception, spatial disorientation, and illusions. NASA CR-1309, 1969

[3] Melvill Jones G: Origin significance and amelioration of Coriolis illusions from the semicircular canals: A non-mathematical appraisal. Aerospace Med 41: 483-490,1970

[4] Guedry FE Jr, Benson AJ: Coriolis cross-coupling effects: Disorienting and nauseogenic or not? Aviat Space Environ Med 49: 29-35,1978

[5] Naoki Isu:Equilibrium Res 58 : 632-639,1999(in Japanese)

[6] Naoki Isu:Equilibrium Res 59 : 29-37,2000(in Japanese)

[7] Isu N, Shimizu T, and Sugata K: Mechanics of Coriolis stimulus and inducing factors of motion sickness. Biol Sci Space, 15: 415-420, 2001

[8] Blanks RHI, Curthoys IS, Markham CH: Palnar relationships of the semicircular canals in man. Acta Otolaryngol 80: 185-195, 1975

[9] Curthoys IS, Blanks RHI, Markham CH: Semicircular canal functional anatomy in cat, guinea pig and man. Act Otolaryngol 83: 258-265, 1977

[10] Grunfeld EA, Okada T, Jauregui-Renaud K, et al.; The effect of habituation and plane of rotation on vestibular perceptual responses. J Vestib Res, 10: 193-200,2000

[11] Okada T, Grunfeld EA, Shallo-Hoffmann J, et al. ; Vestibular perception of angular velocity in normal subjects and in patients with congenital nystagmus. Brain, 122: 1293-1303, 1999

[12] Tweed D, Fetter M, Sievering D, el al: Rotational kinematics of the human vestibuloocular reflex. II . Velocity steps. J Neurophysiol, 72: 2480-2489, 1994

[13] Seidman SH,Leigh RJ: The human torsional vestibulo-ocular reflex during rotation about an earth-vertical axis. Brain Res, 504: 264-268, 1989

[14] Wilson VJ, Melvill Jones G: Mammalian Vestibular Physiology. Pp41-76, Plenum Press, New York, 1979

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Flight control of insects

Toshiatsu Ohmi (Mr)
Dept. of Mechanical Engineering and Intelligent Systems,
The Univ. of Electro-Communications
(ohmi-t@kuroda.mce.uec.ac.jp)

Shigeaki Kuroda (Prof.)
Dept. of Mechanical Engineering and Intelligent Systems,
The Univ. of Electro-Communications
(kuroda@.mce.uec.ac.jp)

Futoshi Tanaka (Dr)
Tokyo University of Science

Kazuhiro Hirasawa (Mr)
Fluid Technology Co, Ltd.

Abstract

In this paper, we proposed a new approach to elucidate free flight of insects. We simulated a free flight of insects, which was quite difficult to elucidate by researches based on observations, and evaluated the results. A free flight of two-dimension and two-degree of freedom of fly *Drosophila* was simulated. The flight pass and flight velocity of insect in a free flight were calculated by using Newton's equation of motion. The flight of virtual insect controlled by the motion of wings was calculated by using a control theory. Then aerodynamic force generated by a flapping motion of insect was estimated by using the blade-element theory. Based on these theories, flight control of insect was performed from the initial point to the target point. An optimal regulator theory was used as a control law. The optimal regulator theory calculates the optimal control input based on minimum energy condition in a free flight. The trajectory to the target point and motion of wing were obtained by the results of free flight simulation of virtual insects. The results suggested that this approach was a new method to elucidate the free flight of insects.
Key words: flight control, free flight of insects, blade-element theory, optimal regulator theory, simulation.

1. Introduction

Existing various flight insects have survived natural selection for a long time and have accomplished notable evolution in the process. The flight performances of these insects are very acrobatic such that they turn on a pinpoint, fly sideways and even sit upside-down on a ceiling. These flight performances are very attractive to us. These remarkable flight skills are produced by various changes of the motions in a flight insect body. Insects modulate the magnitude of aerodynamic force by changing flapping frequency and amplitude. They change the direction of flight by modulating angle of stroke plane. Various researches based on observations of flight of insects have been made to elucidate free flight of insects. However the experimental researches were mainly done with the tethered insects. This is because it is quite difficult to measure insect's behaviors in free flight. A new approach is proposed to elucidate a free flight of insects in this paper. This new approach uses the results of the free flight simulation of insects. A flapping of wings was controlled based on control theory, and the aerodynamic force generated by an insect was calculated by using aerodynamic theory. The free flight of two-dimension and two-degree of freedom of fly *Drosophila* was simulated from the initial point to the target point by using optimal regulator theory. The wing motion was analyzed in the free flight.

2. Principles

Fig.1 shows schematic of an insect in the free flight. We simulate the free flight of an insect in a stationary air. A flight pass of an insect is fixed to two-dimensional plane, and it is assumed that an insect moves only in the vertical and horizontal directions.

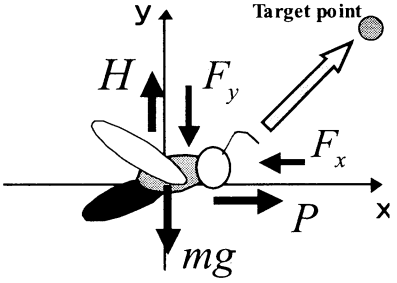


fig. 1 schematic of an insect in a free flight

2.1. Equations of motion

The equations of motion, which determine the position and velocity of an insect, are as follows:

$$m\ddot{y} = -mg + 2H - F_y \quad (1)$$

$$m\ddot{x} = 2P - F_x \quad (2)$$

where, m is the mass of an insect, g is the gravitational acceleration, $2H$ and $2P$ are

the hovering force and the thrust force, which are generated by a pair of wings, respectively. F_y and F_x are aerodynamic drag in the vertical and horizontal directions, acting on the body of an insect.

2.2. Blade element theory

H and P, which are the external forces of the equations of motion, are calculated based on the blade element theory. Drag and lift around the wing element are shown in fig. 2. In the blade element theory, wing is decomposed into a set of blade-elements. Then, aerodynamic force is calculated from the dynamic pressure acting on each wing-element. The aerodynamic force acting on the wing is calculated by integrating the aerodynamic force acting on each wing elements in the span-wise direction. When the blade-element theory is used, the lift and drag force are given as follows:

$$L = \int_0^R \tfrac{1}{2}\rho c V^2 C_L(\alpha)dr \tag{3}$$

$$D = \int_0^R \tfrac{1}{2}\rho c V^2 C_D(\alpha)dr \tag{4}$$

where, ρ is the density of air, c is the wing chord, R is the wing span, α is the angle of attack, V is the relative velocity generated by the combination of flapping motion and the flight speed of an insect. C_D and C_L are drag and lift coefficients respectively. Aerodynamic forces around stroke plane are shown in fig. 3

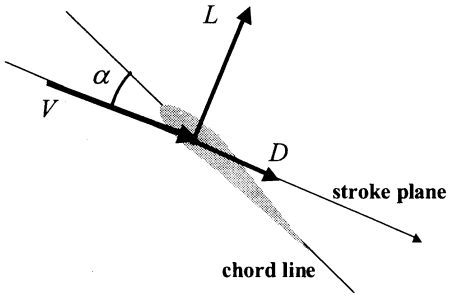


fig. 2 Lift and Drag

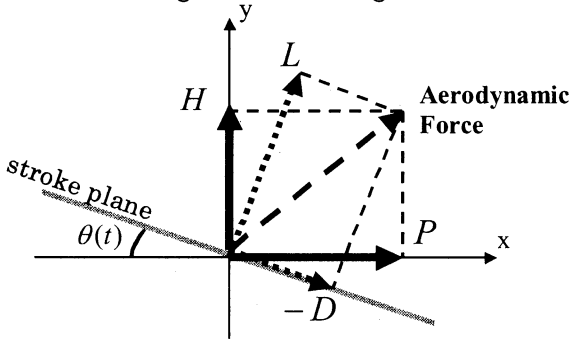


fig. 3 Aerodynamic force

H and P are the averaged value of forces generated in a flapping cycle. H and P are given by:

$$H = \cos\theta(t)\frac{1}{T}\int_0^T\int_0^R \frac{1}{2}\rho c dr V(\tau,r)^2 C_L(\alpha(\tau))d\tau + \sin\theta(t)\frac{1}{T}\int_0^T\int_0^R \frac{1}{2}\rho c dr V(\tau,r)^2 C_D(\alpha(\tau))d\tau \tag{5}$$

$$P = \sin\theta(t)\frac{1}{T}\int_0^T\int_0^R \frac{1}{2}\rho c dr V(\tau,r)^2 C_L(\alpha(\tau))d\tau - \cos\theta(t)\frac{1}{T}\int_0^T\int_0^R \frac{1}{2}\rho c dr V(\tau,r)^2 C_D(\alpha(\tau))d\tau \tag{6}$$

where T is the period of a flapping cycle and $\theta(t)$ is the angle of stroke plane. t is the time. r is the distance from the root of wing to the wing-element. C_D and C_L are drag and lift coefficients. We used the values of C_D and C_L measured by M.H.Dickinson et al[1]. The projection of stroke plane is shown in fig. 4.

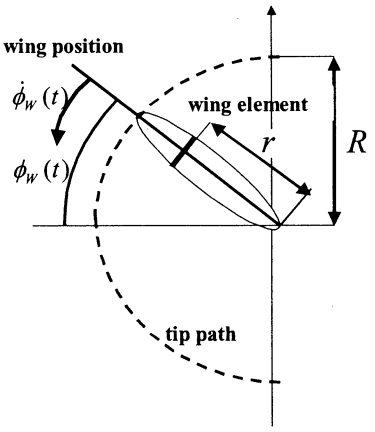


fig. 4 projection of stroke plane

Velocity vectors around a wing element are shown in fig. 5. Airspeed $V(t,r)$ is resultant velocity, which derived from the flapping velocity of wing $V_W(t,r)$ and the flight speed $V_B(t)$. The flight speed $V_B(t)$ is represented by horizontal velocity component $\dot{x}$ and vertical velocity component $\dot{y}$.

$$V_B(t) = \left(\dot{x}^2 + \dot{y}^2\right)^{1/2} \tag{7}$$

Since the flapping velocity of wing $V_W(t,r)$ is very high compared with the flight speed $V_B(t)$, change of the direction of a resultant velocity vector by the flight speed $V_B(t)$ is disregarded. The flapping velocity of wing is given as follows.

$$V_W(t,r) = r\dot{\phi}_W(t) \tag{8}$$

Here $\dot{\phi}_W(t)$ is the flapping angular velocity. The resultant velocity $V(t,r)$ is represented by

using equation (7) and (8) as follows.

$$V(t,r) = \left((r\dot{\phi}_W(t) - \dot{x}\cos\theta(t) + \dot{y}\sin(t))^2 + (\dot{x}\sin\theta(t) + \dot{y}\cos(t))^2\right)^{1/2} \quad (9)$$

The motion of flapping $\phi_W(t)$ and feathering $\alpha(t)$ are represented by using trigonometric functions

$$\phi_W(t) = \phi(t)\sin(2\pi ft) \quad (10)$$

$$\alpha(t) = \alpha_0\cos(2\pi ft) + \pi/2 \quad (11)$$

Here, f is the flapping frequency, $\phi(t)$ is the flapping amplitude, α_0 is the amplitude of the angle of attack.

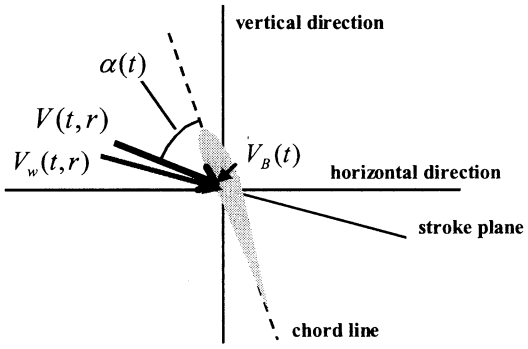


fig. 5 resultant velocity

Aerodynamic drag acting on an insect are given as equations (12) and (13). The body of an insect is approximated as a sphere, and the radius of a sphere is equal to a wingspan.

$$F_y = \frac{1}{2}\rho S(\dot{x}^2 + \dot{y}^2)C_f\cos\beta \quad (12)$$

$$F_x = \frac{1}{2}\rho S(\dot{x}^2 + \dot{y}^2)C_f\sin\beta \quad (13)$$

$$\beta = Tan^{-1}\frac{\dot{y}}{\dot{x}} \quad S = \pi R^2$$

Here, S is projection area and C_f is resistance coefficient. The resistance coefficient C_f was set to 1.0.

2.3. Control theory

It is assumed that insects control the angle of stroke plane $\theta(t)$ and the flapping amplitude $\phi(t)$ for flight control. The angle of stroke plane $\theta(t)$ modulates the direction of aerodynamic force and the flapping amplitude $\phi(t)$ modulates the magnitude of aerodynamic force.

2.3.1. Linearization of equations

The equations (1) and (2) are transformed into the linear equations. The unknown variables are represented as the sum of the equilibrium values and the fluctuations from the equilibrium values. Those equations are approximated with Taylor expansion to obtain the linearized equations (14) and (15).

$$m\ddot{y}(t) = \tilde{\phi}(t)(2A_1D_1\bar{\phi} + A_2D_3E_1) + \tilde{\theta}(t)(A_1D_2\bar{\phi}^2 + A_2(-D_3E_2 + D_4E_1)\bar{\phi} + A_3D_6(E_1^2 + E_2^2)) \quad (14)$$

$$m\ddot{x}(t) = \tilde{\phi}(t)(-2A_1D_2\bar{\phi} - A_2D_4E_1) + \tilde{\theta}(t)(A_1D_1\bar{\phi}^2 + A_2(D_3E_1 + D_4E_2)\bar{\phi} + A_3D_5(E_1^2 + E_2^2)) \quad (15)$$

$$A_1 = \frac{4}{3}\rho c\pi^2 f^2 R^3,\ A_2 = 2\rho c\pi fR^2,\ A_3 = \rho cR$$

$$D_1 = I_{L1}\cos\bar{\theta} + I_{D1}\sin\bar{\theta} \quad D_2 = -I_{L1}\sin\bar{\theta} + I_{D1}\cos\bar{\theta}$$

$$D_3 = I_{L2}\cos\bar{\theta} + I_{D2}\sin\bar{\theta} \quad D_4 = -I_{L2}\sin\bar{\theta} + I_{D2}\cos\bar{\theta}$$

$$D_5 = I_{L3}\cos\bar{\theta} + I_{D3}\sin\bar{\theta} \quad D_6 = -I_{L3}\sin\bar{\theta} + I_{D3}\cos\bar{\theta}$$

$$E_1 = \dot{x}\cos\bar{\theta} - \dot{y}\sin\bar{\theta} \quad E_2 = \dot{x}\sin\bar{\theta} + \dot{y}\cos\bar{\theta}$$

$$I_{L1} = \frac{1}{T}\int_0^T \cos^2(2\pi f\tau)C_L(\alpha(\tau))d\tau$$

$$I_{L2} = \frac{1}{T}\int_0^T \cos(2\pi f\tau)C_L(\alpha(\tau))d\tau \quad I_{L3} = \frac{1}{T}\int_0^T C_L(\alpha(\tau))d\tau$$

$$I_{D1} = \frac{1}{T}\int_0^T \cos^2(2\pi f\tau)C_D(\alpha(\tau))d\tau$$

$$I_{D2} = \frac{1}{T}\int_0^T \cos(2\pi f\tau)C_D(\alpha(\tau))d\tau \quad I_{D3} = \frac{1}{T}\int_0^T C_D(\alpha(\tau))d\tau$$

$$F_1 = \frac{1}{2}\rho\pi R^2(\dot{x}^2 + \dot{y}^2)C_f\sin\beta$$

$$F_2 = \frac{1}{2}\rho\pi R^2(\dot{x}^2 + \dot{y}^2)C_f\cos\beta$$

Here, $\bar{\theta}$ and $\bar{\phi}$ are the angle of stroke plane and flapping amplitude of the equilibrium point. $\tilde{\theta}(t)$ and $\tilde{\phi}(t)$ are the fluctuations from the equilibrium values of the angle of stroke plane and the flapping amplitude. The values of equilibrium point are estimated by following equations.

$$0 = -mg + A_1D_1\bar{\phi}^2 + A_2D_3E_1\bar{\phi} + A_3D_5(E_1^2 + E_2^2) - F_1 \quad (16)$$

$$0 = -A_1D_2\bar{\phi}^2 - A_2D_4E_1\bar{\phi} - A_3D_6(E_1^2 + E_2^2) - F_2 \quad (17)$$

2.3.2. Optimal control

As a result of linearization, the controlled system are express as follows.

$$\dot{X} = AX + BU \quad (18)$$

$$Y = CX \quad (19)$$

X is the state vector ($x_1 = y(t) - y_r$, $x_2 = \dot{y}(t)$, $x_3 = x(t) - x_r$, $x_4 = \dot{x}(t)$), U is the control input ($\tilde{\theta}(t)$, $\tilde{\phi}(t)$) and Y is the output vector($y(t) = x_1 + y_r$, $x(t) = x_3 + x_r$). In order to determine the control inputs required to reach the target point from the initial point, the optimal regulator theory was applied to this linear system. The performance index was applied in a quadratic form as follows.

$$J = \int_0^\infty [X^T QX + U^T RU]dt \qquad (20)$$

The control input U is determined to minimize the performance index J. Q and R are the weighting matrix. Q must be positive definite symmetric matrix. Similarly R must be semi-positive definite symmetric matrix. The optimal state feedback coefficient matrix F is given as follows:

$$F(t) = -R^{-1}B^T P \qquad (21)$$

where P, which is estimated by using Riccati equation, is semi-positive definite symmetric matrix. Riccati equation is given as follows.

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \qquad (22)$$

The state feedback coefficient matrix F is transformed into the optimal control input U by using equation (23).

$$U = FX \qquad (23)$$

The optimal control inputs are estimated from the linear equations (18) and (19), these input U is used to calculate the new state vector from the non-linear equations (1) and (2). Time integration of equations (1) and (2) was carried out by using the explicit Euler method.

3. Results and discussion

The computation of the flight trajectory using the above mentioned control method was performed from the initial point to the target point to evaluate the validity of the method. The conditions of calculation and the flight data are shown in table. 1 and table. 2 respectively. The flight data were quoted from the data described in Zanker et al[2] and Weis-Fogh[3][4].

table. 1 the conditions of calculation

initial position(x,y)[m]	(0,0)
initial velocity(u,v)[m]	(0,0)
target point(xr,yr)[m]	(1.0,0.5)
Δt (s)	5.4×10^{-3}

The state variables, the input variables and the output variables were set as follows.

$$x_1 = y(t) - y_r \qquad x_2 = \dot{y}(t) = \dot{x}_1$$
$$x_3 = x(t) - x_r \qquad x_4 = \dot{x}(t) = \dot{x}_3 \qquad (24)$$
$$u_1 = \tilde{\theta}(t) \qquad u_2 = \tilde{\phi}(t)$$

Each matrix of the equations (18) and (19) are shown as follows.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ b_{21} & b_{22} \\ 0 & 0 \\ b_{41} & b_{42} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$b_{12} = \left(A_1 D_2 \bar{\phi}^2 + A_2(-D_3 E_2 + D_4 E_1)\bar{\phi} + A_3 D_6 (E_1^2 + E_2^2)\right)/m$$
$$b_{22} = \left(2A_1 D_1 \bar{\phi} + A_2 D_3 E_1\right)/m$$
$$b_{41} = \left(A_1 D_1 \bar{\phi}^2 + A_2(D_3 E_1 + D_4 E_2)\bar{\phi} + A_3 D_5 (E_1^2 + E_2^2)\right)/m$$
$$b_{42} = \left(-2A_1 D_2 \bar{\phi} - A_2 D_4 E_1\right)/m$$

The weighting matrix Q and R are set as identity matrix.

table. 2 flight data[2],[3],[4]

density(ρ)	1.184(kg/m^3)
flapping frequency (f)	185(Hz)
wing shape	rectangular plate
wing span	2.5×10^{-3}(m)
wing chord	0.7×10^{-3}(m)
feathering amplitude (α o)	70(deg)
mass (m)	2.0×10^{-3}(kg)

fig. 6 shows the flight trajectory from the initial point to the target point. The flight trajectory is linearly directed to target point. The flight moving speed is fast in the early stage.

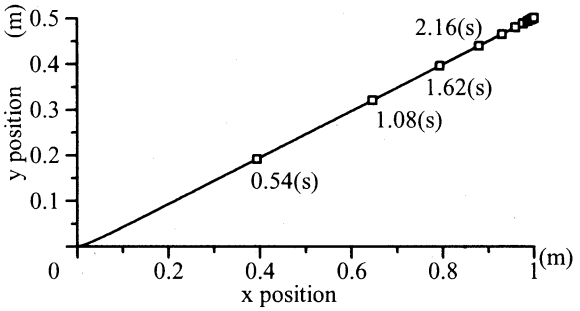


fig. 6 trajectory to target point

The time histories of positions $x(t), y(t)$ and velocities $u(t), v(t)$ of an insect are shown in fig. 7 and fig. 8 respectively. In the early stage, both $u(t)$ and $v(t)$ sharply increase and then gradually decrease after the maximum value. The maximum flight velocity during the flight is 0.95 (m/s). The measured result of the flight velocity by David[5] is up to 1.0 (m/s), and present computational result corresponds to that of David.

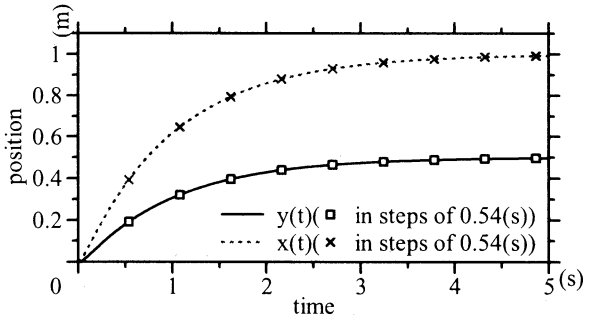


fig. 7 time histories of x(t)-y(t) positions

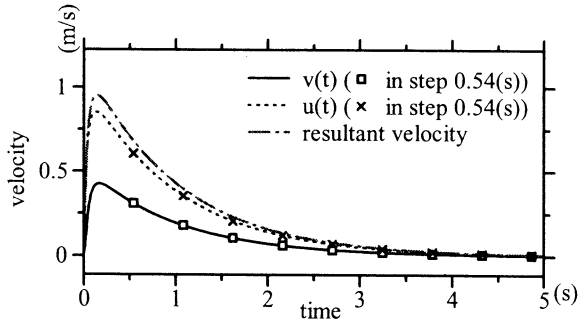


fig. 8 time histories of u(t)-v(t) velocities

The time histories of angle of stroke plane and flapping amplitude are shown in fig. 9. The time histories of hovering force and thrust force are shown in fig. 10. The stroke plane inclines greatly in the early stage. As shown also in fig.10, the aerodynamic force mainly directed horizontally. About 1 second after the beginning of flight, the aerodynamic force in the horizontal direction is almost zero. The insect flies during this period slowing down the flight speed gradually by aerodynamic drag. The flapping amplitude in the hovering flight calculated from present analysis is 47.8 degrees. That data measured by the other researchers[2],[4] ranges from 60(deg) to 80 (deg). The present calculated value is smaller than the measured one. As the lift in the present calculation is overestimated because the wing shape used in this calculation is rectangular plate so that the chord around the wing tip is longer than that of the real wing.

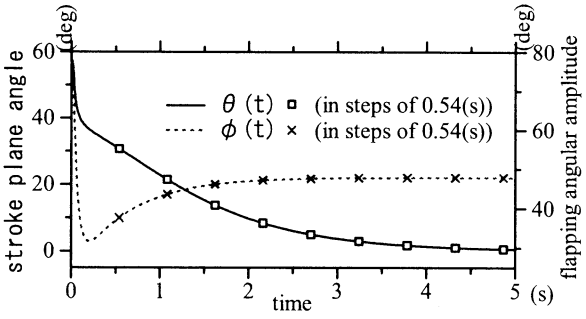


fig. 9 time histories of stroke angle $\tilde{\theta}(t)$ and flapping amplitude $\tilde{\phi}(t)$

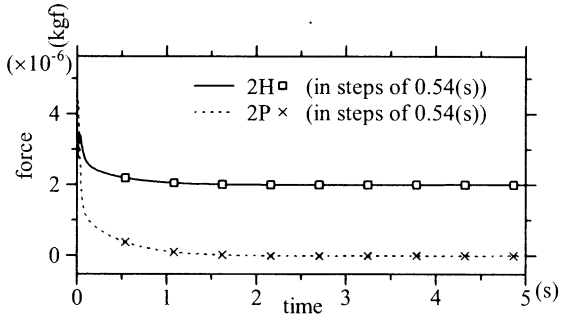


fig. 10 time histories of H and P

4. Conclusion

The flight of the virtual insect was simulated based on minimum energy condition by using the optimal regulator theory, and the trajectory to target point and motion of wing were obtained from the results of free flight simulation of virtual insects. In order to make this simulation more realistic, it is necessary to perform the free flight simulation of three-dimension and six-degree of freedom. Also, evaluation of unsteady aerodynamic force is still inadequate. Although there are subjects which should be solved, it is suggested that the new method will be applied to elucidate free flight of insects.

5. Reference

[1]M.H.Dickinson, Fritz-Olaf, Sanjay P.Sane (1999), Science284, 1954
[2]J.M.Zanker, K.G.Gotz(1990),Phil. Trans. R. Soc. Lond. B 327,pp19-44
[3]Weis-Fogh(1972),J.Exp.Biol,56,pp79-104
[4]Weis-Fogh(1973),J.Exp.Biol,59,pp169-230
[5]David.C.T(1978),Physiol. Ent. 3,pp191-195

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Environment Reconstruction and Navigation of a Mobile Robot Using an Active Vision

Tae-Seok Jin*, JinWoo Park**, and Jang-Myung Lee*

*Dept. of Electronics Engg.
Pusan National University
Pusan, 609-735, Korea
jints@pusan.ac.kr

**Institute of Information
Technology Assessment
jinu@iita.re.kr

Abstract

In this paper, as the preliminary step for developing a multi-purpose autonomous robust carrier mobile robot to transport trolleys or heavy goods and serve as robotic nursing assistant in hospital wards. This paper presents the advanced method of multi-sensor data fusion for map building and mobile robot navigation, and present an experimental mobile robot designed to operate autonomously within both indoor and outdoor environments. We will also describe the on-going experimental work on the Multi-Purpose Autonomous Vehicle platform on local navigation using new multisensor fusion method used ultra-sonar and active vision camera. It is dealt with detailed functions recognizing environments updated, obstacle detection and motion assessment, with the first results from the simulations run.

1. Introduction

Sensing of the environment and subsequent control is important feature of the navigation of an autonomous mobile robot. When a mobile robot navigates in an unknown or partially known environment, several types of sensors are commonly used for this purpose such as ultrasonic sensors, infrared sensors, laser range finders and vision systems for obstacle avoidance or path planning. Recently, it is increasing the use of vision system because it has inexpensive and is able to be fast real-time environmental recognition (Allen *et al.*, 1991; Camillo *et al.*, 1998). In this paper we present a statistical method for dealing with the general problem of concurrent localization and map building. We furthermore address the problem of using occupancy grid maps for path planning in highly dynamic environments. The approaches have been tested extensively and several experimental results are given in the paper. The method is based on a variant of the EM algorithm, which is an efficient hill-climbing method for maximum likelihood estimation in high-dimensional spaces. In the context of mapping, EM iterates two alternating steps: a *localization step*, in which the robot is localized using a previously computed map, and a *mapping step*, which computes the most likely map based on the previous pose estimates.

2. Related Theories

2.1 Occupancy Grids

Occupancy Grids is certainty a state of the art method in the field of grid based methods. The idea is to divide the environment into grid cell C_{ij} . Typically a 2-dimensional grid is enough to give interesting information about the environment. Each cell can be in two states $s(C_{ij}) = \mathrm{OCCupied}$ or $s(C_{ij}) = \mathrm{EMPty}$, and to each cell there is a probability $P[s(C_{ij}) = \mathrm{OCC}]$ attached, which reflects the belief of the cell C_{ij} being occupied by an object.
Since

$$P[s(C_{ij}) = \mathrm{EMP}] = 1 - P[s(C_{ij}) = \mathrm{OCC}] \quad (1)$$

The grid is initialized with $P[s(C_{ij}) = \mathrm{OCC}] = 1/2$. To update the cells when the robot traverses the environment, a stochastic sensor model $p(\mathrm{r} \mid \mathrm{z})$ is used. This model is obtained from experiments with the sensor in question and relates reading vector, r, to the true space range vector, z. Given a new range reading, r, from a sensor, the idea now is to use the sensor model $p(\mathrm{r} \mid \mathrm{z})$ to update the probabilities $P[s(C_{ij}) = \mathrm{OCC}]$ in the Occupancy Grid. This can be done by using Bayes' theorem, see Equation (4),

$$P[s(C_{ij}) = \mathrm{OCC} \mid \mathrm{r}] = \frac{p[\mathrm{r} \mid s(C_{ij}) = \mathrm{OCC}] P[\mathrm{s}(C_{ij}) = \mathrm{OCC}]}{\sum_{s(C_{ij})} p[\mathrm{r} \mid s(C_{ij})] P[s(C_{ij})]} \quad (2)$$

We mention here that the right side of Equation (2) has to

be developed further to be computable. Exactly how this is done can be found in (Abidi *et al.*, 1992; Elfes, 1989). The Occupancy grid method now provides a useful setting for fusing data from different sensors. There are, basically, two main approaches:

2.2 Vector Field Histogram (VFH)

Careful analysis of the shortcomings of the VFF method reveals its inherent problem: excessively drastic data reduction that occurs when the individual repulsive forces from histogram grid cells are added up to calculate the resultant obstacle vector m_{ij}. Hundreds of data points are reduced in one drastic step to only two items: direction and magnitude of m_{ij}. Consequently, detailed information about the local obstacle distribution is lost. To remedy this shortcoming, we have used a new method called the *vector field histogram* (VFH). A window moves with the robot, overlying a square region of active window in the histogram grid of Figure 1.

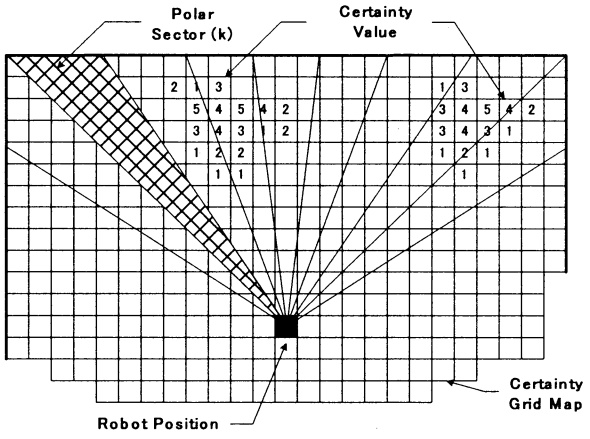


Fig. 1. The heart of the VFH method: Mapping active cells onto the polar histogram.

The contents of each active cell in the histogram grid are mapped into the corresponding sector of the polar histogram (see Fig. 1), resulting in each sector k(S_k) holding a value h_k. Thus, h_k is higher if there are many cells with high certainty value c_{ij} s in one sector. Intuitively, this value can be interpreted as the polar obstacle density in the direction of sector k(S_k). For each cell C_{ij} in a given sector, say S_k, one calculates an obstacle vector m_{ij}. The magnitude of m_{ij} is dependent of the certainty value c_{ij}, and also by the distance between the center of the grid (robot position) and the cell C_{ij}. After this procedure one sum all the obstacle vectors m_{ij} s in sector S_k to form an obstacle density entity h_k

$$h_k = \sum_{C_{ij} \in S_k} m_{ij}, \qquad k = 1, \ldots n. \tag{3}$$

Here, Robot fixed certainty grid with certainty value c_{ij}. The map is divided into polar sectors.

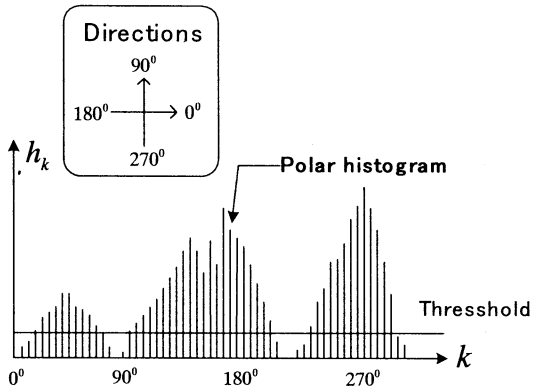


Figure 2. The Polar Histogram.

Figure 2 shows the polar histogram corresponding to the momentary position of the mobile robot. The directions in the polar histogram correspond to directions measured counterclockwise from the positive x-axis of the histogram grid. By applying threshold one can determine sector regions where it is possible to do obstacle avoidance.

3. Data Fusion

3.1 Statistical Foundations

From a statistical perspective, we have the following problem. Given two vector random variables X and Y, what does the observation $Y = y$ tell us about X? The complete answer is given by the so-called conditional probability density function,

$$p_{X|Y}(x \mid y) = \frac{p_{X|Y}(x,y)}{p_Y(y)} \tag{4}$$

Here $p_{X,Y}(x,y)$ is the joint probability density for X and Y, and $p_Y(y)$ is the probability density for Y. By using the dual assumption, namely that $X = x$ is given, we obtained the very useful Bayes rule

$$p_{X,Y}(x,y) = p_{X|Y}(x \mid y)p_Y(y) = p_{Y|X}(y \mid x)p_x(x) \tag{5}$$

$$p_{X,Y}(x,y) = \frac{p_{Y|X}(y \mid x)p_X(x)}{p_Y(y)}, \tag{6}$$

which is the key formula in Bayesian and maximum likelihood estimation theory.

Different estimates of X can now be constructed from its distribution. The (conditional) minimal variance of X equals the conditional mean of X given $Y = y$,

$$\hat{x} = E[X \mid Y = y] = \int_{-\infty}^{\infty} x p_{X|Y}(x \mid y) dx \qquad (7)$$

Another useful estimate is the maximum a posteriori estimate, which maximizes the function $p_{X|Y}(x \mid y)$. The rest is design and analysis issues, i.e. formulating the underlying model, specifying probability density functions and calculating equality/variance properties. The most used probability density function is the Gaussian one (the Normal distribution). The main reason is that the conditional density function also will be Gaussian, and analytic expressions of the minimal variance estimate can thus be obtained.
Let X and Y be jointly Gaussian, i.e. $Z = [X' Y']'$ is Gaussian with mean and covariance

$$m_z = \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}, \quad \Sigma_{zz} = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \qquad (8)$$

Then X conditional on $Y = y$ has a Gaussian distribution with mean and covariance

$$\begin{aligned} m_{x|y} &= \bar{x} + \Sigma_{xy} \Sigma_{yy}^{-1} (y - \bar{y}), \\ \Sigma_{x|y} &= \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx}. \end{aligned} \qquad (9)$$

Hence the conditional mean of X given $Y = y$, equals

$$\hat{x} = E[X \mid Y = y] = \bar{x} + \Sigma_{xy} \Sigma_{yy}^{-1} (y - \bar{y}). \qquad (10)$$

Almost all practical estimators are special cases of the above result. The expression is called *the fundamental equations of linear estimation* in (Bar-Shalom, 1993). This reference also provides a very good introduction to estimation theory, in general, and tracking, in particular.

4. Building Navigation Probability Grids

4.1 Integrating multiple sensor readings

In order to give reliable position estimates we have to integrate the information of consecutive sensor readings. Suppose m is the model of the environment, and $p(x \mid s_1 \wedge \ldots \wedge s_{n-1} \wedge m)$ is the (posterior) probability that x refers to the current position of the robot, given m and the sensor readings $s_1, \ldots, s_{n-1}$. Then, to update the probability for x given new sensory input s_n we use the following update formula (Pearl 1988]:

$$\begin{aligned} &p(x \mid s_1 \wedge \ldots \wedge s_{n-1} \wedge s_n \wedge m) = \\ &\alpha \cdot p(x \mid s_1 \wedge \ldots \wedge s_{n-1} \wedge m) \cdot p(s_n \mid x \wedge m) \end{aligned} \qquad (15)$$

The term $p(s_n \mid x \wedge m)$ is the likelihood of measuring the sensory input s_n given the world model m and assuming that x refers to the current position of the robot. The constant α simply normalizes the sum of the position probabilities over all x up to 1.
To initialize the grid we use the a priori probability $p(x \mid m)$ of x referring to the actual position of the robot given m. The estimation of $p(x \mid m)$ and $p(s_n \mid x \wedge m)$ depends on the given world model and the type of the sensors used for position estimation.

4.2 *Integrating the movements of the mobile robot*

In order to update the grid according to the robot's movements and to deal with possible dead reckoning errors we use a general formula coming from the domain of Markov chains. We regard each cell in P as one possible state of the robot, and determine a state transition probability $p(x \mid \tilde{x} \wedge \tau \wedge t)$ for each pair x, $\tilde{x}$ of cells in P, which depends on the trajectory τ taken by the robot and the time t elapsed since the previous update. This transition probability should also model how the trajectory τ fits into the environment.

$$P[x] := \alpha \cdot \sum_{\tilde{x} \in P} P[\tilde{x}] \cdot p(x \mid \tilde{x} \wedge \tau \wedge t) \qquad (16)$$

where a is a normalizing constant. At any time the field of P with the highest probability represents the best estimation for the current position of the robot. The confidence in this estimation depends on the absolute value of the probability and on the difference to the probabilities in the remaining fields of the grid.

5. Position Estimation with Occupancy Probability Map

5.1 Matching sonar sensor readings against occupancy grids

To compute the likelihood $p(s \mid x, m)$ that a sensor reading s is received given the position x and an occupancy grid map m we use a similar approach as described in (Moravec, 1988). We consider a discretization $R_1, \ldots, R_k$ of possible distances measured by the sensor. Consequently, $p(R_i \mid x \wedge m)$

is the likelihood that the sonar beam is reflected in R_i. Suppose $p(r(\bar{x})|x \wedge m)$ is the likelihood that the cell $\bar{x}$ reflects a sonar beam, given the position x of the robot and the map m. Furthermore suppose that $\bar{x}$ belongs to R_i, Assuming the reflection of the sonar beam by $\tilde{x}$ being conditionally independent of the reflection of the other cells in R_i, the likelihood that R_i reflects a sonar beam is

$$p(R_i|x \wedge m) = 1 - \prod_{\tilde{x} \in R_i} (1 - p(r(\tilde{x})|x \wedge m)) \quad (17)$$

Before the beam reaches R_i, it traverses $R_1, \ldots, R_{i-1}$. Supposed that the sonar reading s is included by range R_i, the likelihood $p(s|x \wedge m)$ equals the likelihood that R_i reflects the sonar beam given that none of the ranges $R_{<i}$ reflects it. Thus, we have

$$p(R_i|x \wedge m) \cdot \prod_{j=1}^{i-1} (1 - p(R_j|x \wedge m)) \quad (18)$$

In Computing position estimates using occupancy grids. It remains to estimate the initial probability $p(s|m)$ that the field x of m contains the current position of the robot. We assume that this probability directly depends on the occupancy probability $m(x)$ of the field x in m: the higher the occupancy probability, the lower is the position probability and vice versa. Therefore, the value $p(s|m)$ is computed as follows:

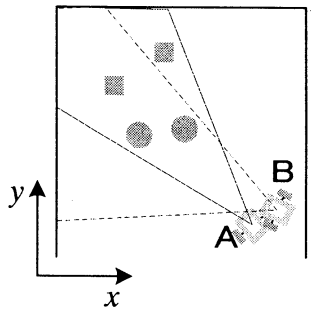
$$p(s|m) = \frac{1 - m(x)}{\sum_{\tilde{x} \in m} (1 - m(\tilde{x}))} \quad (19)$$

5.2 Implementation aspects

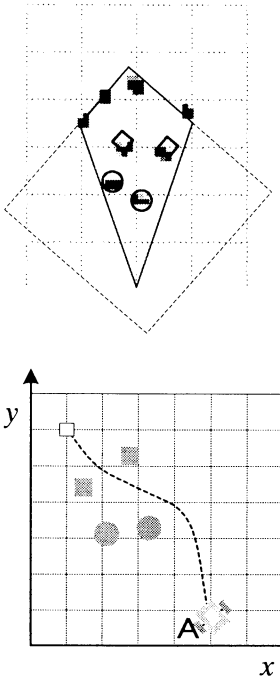
For the sake of efficiency we implemented a simplified model of sonar sensors to compute the likelihood of a reading: instead of considering all cells of the grid covered by the sonar wedge as done in (Moravec 1988) we only consider the cells on the acoustic axis of the sensor.

6. Experimental Results

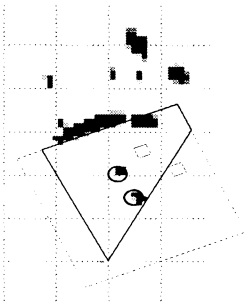
We executed the experiments that a mobile robot navigate to the target point having the initial condition in figure 3 using NOMAD platform . We fixed initial point and target point of the mobile robot to initial point (x:500cm, y:80cm, 120°) and target point (x:100cm, y=600), and initial point (x:600cm, y:100cm, 150°) and target point (x:100cm, y=600). The results is showed in figure 3(b) and 3(c). Probability method that we proposed navigation algorithm[15] is used for map building and navigation. The fusion formula just means that estimates should be weighted together, with weights inversely proportional to their qualities/variances.

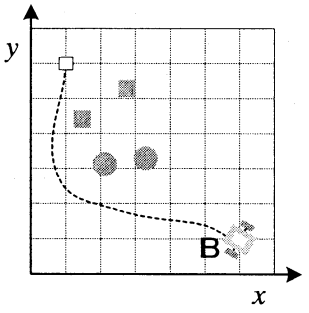


(a) Diagram of navigation environment



(b) Map building and navigation [A]: Initial point (500,80,120°), Target point (100,600)





(c) Map building and navigation [B]: Initial point (600,100,150°), Target point (100,600)

Figure 3. Experimental results of map building and navigation

Figure 3(b) showed that the map-building and navigation is completed successfully. In case of figure 3(c), NOMAD platform navigated using map information and finished safely, but map- building wasn't correct slightly. It results from the error of recogniton. Because image information from CCD camera based on 2D coordinate system isn't sufficient to compute compont in Z plane although obstacles is seperated each in real environment. However, through many experiments, information obtained from the proposed method was suffiicient to build a map and navigate a complete and reliable representation.

7. Conclusion

We have presented a method for building maintaining sensory probabilistic maps and navigation that can be used for mobile robot navigation and localization in unknown environment. We modeled uncertainties of a multi-sensor system and proposed a method of making probability map of sensor data information with uncertainties for navigation. To show the efficiency of proposed method, real experiments are preformed. The experimental results show that a mobile robot can construct a map and can navigate to a target point safely according to the map under the unknown environment. We furthermore showed how to apply occupancy grid maps to path planning in highly dynamic environments.
The key idea of this approach is to treat static and dynamic obstacles differently by representing them in different maps. Thereby we decouple the problems of estimating the position of the robot and estimating the positions of static/dynamic obstacles in the environment. However, the image processing takes long (A frame of image takes about 1 second to be processed). Therefore, it needs to study to decrease the image-processing time.

As a future work, we suggest that an active camera system is suitable for tracking moving objects since it can be placed as we want. Results of experiments demonstrated the application of the described methods to a real mobile robot.

References

[1] A. Elfes, *"A Probabilistic Framework for Robot Perception and Navigation,* PhD these, Carnegie-Mellon University, 1989.

[2] A. Elfes "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, RA-3(3): pp. 249-265, 1987.

[3] Allen P., Yoshimi B., and Timecenko A. "Real-time visual servoing," *in Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 851-856, 1991.

[4] Borenstein, J., and Koren, Y. "The vector field histogram - fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation* 7(3): pp. 278–288, 1991.

[5] B. Triggs and C. Laugier "Automatic Camera Placement for Robot Vision," *Proc. IEEE Int'l Conf. Robotics and Automation*, Vol. 2, pp. 1,732-1,738, Nagoya, Japan, 1995.

[6] Buhmann, J., Burgard, W., Cremers, A. B., Fox, D., Hofmann, T., Schneider, F., Strikos, J., and Thrun, S. "The mobile robot Rhino" *AI Magazine* 16(2), pp. 31–38, 1995.

[7] Camillo J. T. and David J. K. "Vision-Based Motion Planing and Exploration Algorithms for Mobile Robots," *IEEE Trans. on Robotics and Automation*, Vol.14, No.3, pp. 417-426, 1998.

[8] D. Fox. *Markov Localization "A Probabilistic Framework for Mobile Robot Localization and Naviagation*." PhD thesis, Dept. of Computer Science, University of Bonn, Germany, December, 1998.

[9] D. Fox, W. Burgard, S. Thrun, and A.B. Cremers "Position estimation for mobile robots in dynamic environments," In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 1998.

[10] Eric M. and Francois C. "Active Vision for Complete Scene Reconstruction and Exploration," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.21, No.1, pp. 65-72, 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Trajectory Estimation of a Moving Object using Kohonen Networks

SangJoo Kim, JaeHo Lee, and JangMyung Lee

Department of Electronics Engineering, Pusan National University
San 30 Jangjeon-dong Kumjeong-ku, Busan, 609-735, Korea
jmlee@pusan.ac.kr

Abstract

A novel approach to estimate the real time moving trajectory of an object is proposed in this paper. The object position is obtained from the image data of a CCD camera, while a state estimator predicts the linear and angular velocities of the moving object. To overcome the uncertainties and noises residing in the input data, a Kalman filter and neural networks are utilized. Since the Kalman filter needs to approximate a non-linear system into a linear model to estimate the states, there always exist errors as well as uncertainties again. To resolve this problem, the neural networks are adopted in this approach, which have high adaptability with the memory of the input-output relationship. Kohonen Network(Self-Organized Map) is selected to learn the motion trajectory since it is spatially oriented. The superiority of the proposed algorithm is demonstrated through the real experiments.

1. Introduction

Detection of moving objects is encountered in industrial robotic systems, recognition, monitoring and unmanned systems [1-2]. Prediction of the trajectory of moving objects is required for the servo system that aims at control and observation of motion information such as object position, velocity and acceleration and is required for the industrial robots. For a simple example, in the pick and place operation with a manipulator, the precise motion estimation of the object on the conveyor belt is a critical factor for the successful operation of the stable grasping. The well-structured environment such as the moving-jig that carries the object on the conveyor belt and stops when the manipulator grasps the object may release the motion estimation requirement. However, the well-structured environment limits the flexibility of the production line, requires skillful designers for the jig, costs high maintenance expense and eventually it will be disappeared from the automated production line.

To overcome these problems—to grasp a moving object stably without stopping the motion, the trajectory prediction of the moving object on the conveyor belt is necessary. The manipulator control system needs to estimate the highest accuracy position, velocity, and acceleration at any instance to capture the moving object without collision safely and to lift up the object without slippage stably.

When the motion trajectory is not high random and continuous, the trajectory can be model analytically to predict the near future values based upon the measured previous data [3]. However, this kind of approach requires high computational time for the high degrees of freedom motion and its computational complexity increases rapidly when the modeling errors become high. In addition to these, the performance is highly sensitive to the changes of the environment.

In this paper, a novel approach for the real time trajectory estimation of a moving object is proposed. For the image data capturing, a CCD camera is utilized. Through the geometrical analysis of the camera and the object, the position of the object can be estimated [2]. There are several approaches where the state estimator is used to predict the linear and angular velocities.

The most general approach known so far is the Kalman filter whose performance is well verified by the numerous researches [4-8]. However, the Kalman filter is not properly applied for the noisy environment. The adaptive or extended Kalman filter is proposed to improve the prediction accuracy [9]. To make the system robust against the noises in the input data and uncertainties, in this approach, neural networks are suitably incorporated into the Kalman filter. That is when the Kalman filter cannot predict the highly nonlinear trajectories, the estimation role is passed to the neural networks which are trained previously for the nonlinear properties. Since the neural networks are trained only by the relationship between the input and output, this approach is expected to have higher flexibility than the adaptive or extended Kalman filter.

Neural networks can be classified into two categories: supervised learning and unsupervised learning methods. In most of the previous researches, the supervised learning is adopted to overcome the nonlinear properties [10-12]. Since the supervised learning requires the relation of input and output [9], it is not suitable for the real time trajectory estimation where the input-output relation cannot be obtained in the unstructured environment. Therefore, in this paper, SOM(Self Organizing Map) that is a sort of the unsupervised learning is selected to estimate the highly nonlinear trajectory that cannot be predicted by the Kalman filter properly. Fig. 1 summarizes the trajectory estimation system for this research.

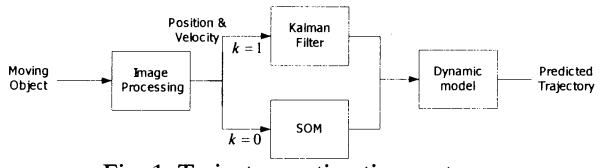


Fig. 1. Trajectory estimation system.

The input for the dynamic model comes from either Kalman filter or SOM according to the following decision equation:

$$predicted\ value = k \cdot Kalman_{out} + (1-k) \cdot SOM_{out} \quad (1)$$

where k=1 for $error \le threshold$ and k=0 for $error > threshold$. The threshold value is determined

empirically based on the size of estimated position error.

2. Parameter learning

2.1 Pre-processing

To classify the moving object pattern in the dynamically changing unstructured environment is not tackled yet successfully [13]. Therefore, for this research, the camera is fixed on the stable basis. To estimate the states of the motion characteristics, the trajectory of the moving object is pre-recorded and analyzed.

As it is recognized in the images, most parts of the CCD image correspond to the background. After eliminating the background, the difference between the two consecutive images frames can be obtained to estimate the moving object motion. To compute the difference either the absolute values of the two image frames or the signed values can be used. This difference method is popular in the image pre-processing to extract desired information from the whole image frame [14].

2.2 Motion vector estimation

Motion vectors can be defined for the moving object at any instance. To predict the motion trajectory using the Kalman filter and SOM, these motion vectors are the basic inputs that should be obtained from the difference images. The motion vector can be computed based on the data, the difference vector from the consecutive two image frames and the sampling period. When the SOM is utilized for the estimation of the motion vector, motion vectors with high correlation resides close to each other, which will be explained in section 4. The fact that motion vectors of neighboring states have high spatial correlation is utilized for estimating the current motion vector in the motion vector estimation process.

3. Kalman Filter

3.1 Modeling of a moving object

When the velocity and acceleration of a moving object are given for the current instance, the position for the next instance, (P_x, P_y), in the Cartesian space can be estimated as follows [15].

$$\hat{P}_{x+\Delta t} = \hat{P}_x + \hat{V}_x \Delta t + \frac{1}{2} A_x \Delta t^2 \quad \text{(2-a)}$$

$$\hat{P}_{y+\Delta t} = \hat{P}_y + \hat{V}_y \Delta t + \frac{1}{2} A_y \Delta t^2 \quad \text{(2-b)}$$

where Δt is the sampling period, and $(\hat{P}_x, \hat{P}_y)$, $(\hat{V}_x, \hat{V}_y)$, and $(\hat{A}_x, \hat{A}_y)$ represent the estimated position, velocity, and acceleration of the moving object, respectively.

An object motion on the xy-plane can be modeled as discrete time varying equations [16] by dividing into linear velocity component, v_k, and angular velocity component, ω_k. The next step position information can be predicted based on the current motion information, v_k and ω_k, as follows:

$$\Delta x_{k+\Delta t,k} = v_k \Delta t \cos(\theta + \frac{1}{2}\omega_k \Delta t) \approx v_k \Delta t \cos\theta_k - \frac{1}{2} v_k \omega_k \Delta t^2 \sin\theta_k \quad (3)$$

$$\Delta y_{k+\Delta t,k} = v_k \Delta t \sin(\theta + \frac{1}{2}\omega_k \Delta t) \approx v_k \Delta t \sin\theta_k - \frac{1}{2} v_k \omega_k \Delta t^2 \cos\theta_k \quad (4)$$

$$\Delta\theta_{k+\Delta t,k} = \omega_k \Delta t \quad (5)$$

where θ_k represents the direction of the moving object *w.r.t* the x-axis on the xy-plane. Also the noises and uncertainties in measuring the velocities, v_k and ω_k, are modeled as follows:

$$\Delta v_{k+\Delta t,k} = \xi_v \quad (6)$$

$$\Delta \omega_{k+\Delta t,k} = \xi_\omega \quad (7)$$

where ξ_v, ξ_ω are zero-mean Gaussian random variables.

To apply a model of the moving object to the Kalman, the previous motion and measurement characteristics need to be converted to a discrete state transition and the observation models that are the followings

$$\begin{aligned} \mathbf{x}_k &= \mathbf{A}_{k,k-1}\mathbf{x}_{k-1} + \xi_{k-1} \\ \mathbf{y}_k &= \mathbf{C}_k x_k + \gamma_k \end{aligned} \quad (8)$$

where $\mathbf{y}_k$ is a measurement vector, $\mathbf{A}_k$ is the state transition matrix, $\mathbf{C}_k$ is the observation matrix, ξ_k represents the irregular components in the state transition, γ_k represents measurement noises, and Δt represents the sampling period.

3.2 State estimation by the Kalman filter

The input data, image data, includes uncertainties and noises during the pre-processing. Therefore, the Kalman filter is suitable for the observer that estimates the states under the noisy environment since the state transition matrix itself has irregular components [4-5][17-18].

In obtaining the filter gain, a covariance matrix for the estimation error is requested.

$$\mathbf{P}'_k = \mathbf{A}_{k,k-1}\mathbf{P}_{k-1}\mathbf{A}^T_{k,k-1} + \mathbf{Q}_{k-1} \quad (9)$$

where $\mathbf{Q}_{k-1}$ is measurement noise covariance matrix. Now the optimal filter gain, $\mathbf{K}_k$, to minimize the state estimation error can be obtained as

$$\mathbf{K}_k = \mathbf{P}'_k\mathbf{C}^T_k[\mathbf{C}_k\mathbf{P}'_k\mathbf{C}^T_k + \mathbf{R}_k]^{-1} \quad (10)$$

where $\mathbf{P}'_k$ is the covariance matrix for the estimation error, $\mathbf{C}_k$ is the observation matrix, and $\mathbf{R}_k$ represents the covariance matrix for the measurement noises. The states are estimated by the following state transition equation where an innovation term is added as an input and multiplied by the Kalman filter gain, $\mathbf{K}_k$, which is the difference between measurement vector, $\mathbf{y}_k$, and the estimated output using the data from the previous step.

$$\hat{x}_k = \mathbf{A}_{k,k-1}\hat{x}_{k-1} + \mathbf{K}_k[\mathbf{y}_k - \mathbf{C}_k\mathbf{A}_{k,k-1}\hat{x}_{k-1}] \quad (13)$$

Before going back to the Eq. (11) for the next step, the covariance matrix of the estimated error needs to be modified as [6-7],

$$\mathbf{P}_k = \mathbf{P}'_k - \mathbf{K}_k \mathbf{C}_k \mathbf{P}'_k \tag{14}$$

4. Self-Organized Map

For a non-linear system, the Kalman filter requires a process to approximate to a quasi-linear model to derive the filtering equations, which leads to the high estimation errors. For the linear model derivation, Taylor series expansion is usually adopted to select the number for terms or to select the order of computational complexity that is inversely proportional to the modeling accuracy. Therefore, it suffers from the trade-off between accuracy and complexity in obtaining a linear model and in estimation the state variables. Especially the Kalman filtering is based on the first order approximation, it neither estimates the states properly all the times nor guarantees the convergence of the states. The adaptive or extended Kalman filter is proposed to overcome this difficulty, which again suffers from the computational complexity too high for real time control. To avoid all these difficult mathematics, there are several ideas for estimating the states using the neural networks [10-12]. In these approaches, they are using the supervised learning schemes that require the data on the pair of the input and output for learning. When the experimental environments are changing dynamically, the supervised learning scheme is not suitable any further since the input and output data sets cannot be utilized efficiently. Therefore, in this approach, SOM as a kind of unsupervised learning scheme is adopted to estimate the trajectory of a moving object.

SOM[19] which known also as Kohonen network has high spatial correlation in the two dimensional space where a moving object is randomly changing its trajectory, which provides the rational for the reason why the SOM is selected among the various unsupervised learning schemes. In the optimal solution search problems, to save the time the data sets or regions are divided into several categories before the search process. For the data division process, VQ (Vector Quantization), SOM, LVQ (Learning Vector Quantization) and other algorithms have been proposed. LVQ algorithm is only effective with the supervised learning, while VQ and SOM are useful for unsupervised learning. However, LVQ is limited to the case of 0 neighborhood. Therefore, SOM becomes the only candidate for our purpose to replace the Kalman filter at the high nonlinear region.

SOM is the most popular unsupervised learning algorithm among the numerous networks to classify and estimate the states and it has the winner-take-all strategy through the competitive learning. SOM has been utilized for image segmentation, object recognition, speech recognition, and vector quantization fields widely.

Each neuron in the SOM calculates and keeps the Euclidian distance that represents the closeness of the connection weight vector and the input vector. The connection weight between the winner neuron j and the neighboring neurons is adaptively changed by

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(x_i(t) - w_{ij}(t))$$

where the parameter, α, is pre-determined for the SOM. SOM is utilized to estimate the position and velocity of the moving object in this paper. During the learning process, SOM determines the optimal states based on the various measurable states. The best estimated set tends sto be selected as a winner neuron. Therefore the stochastic information in Eq. (8) for the Kalman filter is not necessary for the SOM, which enables to use the SOM instead of the Kalman filter for the high nonlinear region where noise distributions and uncertainties are poorly modeled statistically.

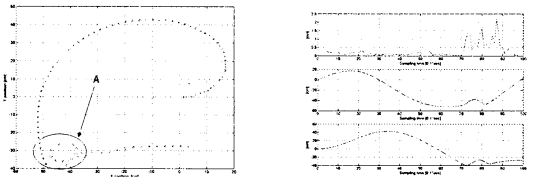
5. Experiments

A micro-mouse is designed for the moving object using the microprocessor, 80C196KC, which generates a non-programmed trace with the maximum speed of 15Cm/sec.

5.1 Trajectory estimation by Kalman filter

Object direction, velocity, and acceleration are obtained from the image data, and in the measuring process, the Gaussian noise of zero-mean and variance = 2, is added to the data. Even though there exists noises, the Kalman filter estimates the states relatively precisely.

To show the high nonlinear effect of the trajectory, a new experiment has been performed.



(a) Trajectory (b)Estimation error/ x, y position.

Fig. 2. Nonlinear trajectory estimation.

As it is shown in Fig. 2(a), when the trajectory of the moving object is highly nonlinear, the Kalman filter estimation has high positional errors and cannot follow the trajectory efficiently. As it shown in Fig. 2(b), the position estimation error is high at the region "A". Therefore, even the system stability cannot be guaranteed for the high nonlinear region.

5.2. Compensation by neural networks

Some experiments are performed to check the superiority of the unsupervised learning to the supervised learning [11-12], and the results are summarized in the table 1.

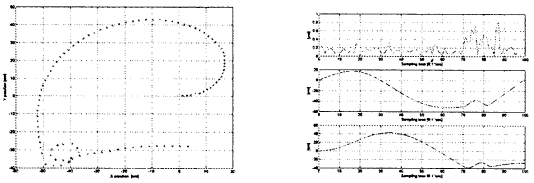
Table 1. Performance comparison of two different neural networks.

Classification / Class	Number of trial	Number of success
Supervised Learning	30	22
Unsupervised Learning	30	27

As it is shown in table 1, even though the supervised learning has also outstanding performance, the

performance degradation becomes severe and becomes unreliable for the dynamically changing environment while the unsupervised learning is consistent. Based on this observation, the SOM is assumed to be the best alternative of the Kalman filter for the nonlinear region.

Another experiment is performed with the SOM instead of the Kalman filter at the nonlinear region to show the superiority of the unsupervised learning scheme, SOM. For the classification of the nonlinear region, the threshold value is empirically adjusted based on the positional estimation error of the Kalman filter.



(a) Trajectory (b)Estimation error/ x, y position.

Fig. 3. Trajectory estimation by SOM.

Fig. 3 shows the experimental results of state estimation by the Kalman filter and the SOM. We need to focus on the nonlinear region where the SOM is applied for the estimation instead of the Kalman filter. By comparing with Fig. 2, it is recognized that the estimation error at the nonlinear region has been decreased by 60%.

6. Conclusion

This research proposes a trajectory estimation scheme for a moving object using the images captured by a CCD camera. In the approach, the state estimator has two algorithms: the Kalman filter estimates the states for the linear approximated region and the SOM for the nonlinear region. The decision for the switch over is made based on the size of the position estimation error that becomes low enough for the linear region and high enough for the nonlinear region. The effectiveness and superiority of the proposed algorithm is verified thorough the experimental data and comparison. The adaptability of the algorithm is also observed during the experiments. For the sake of simplicity, his research is limited to the environment of a fixed camera view. However this can be properly expanded to the moving camera environment where the input data may suffer from high noises and uncertainties. As future research works, selection of a precise learning pattern for the SOM to improve the estimation accuracy and recognition ratio and development of an illumination robust image processing algorithm are left.

Reference

[1] Wan C. Kim, Cheol H. Hwang, Su H. Choi, and Jang M. Lee, "Efficient Tracking of a Moving Object Using Optimal Representative Blocks," *ICCAS 2002, International Conference on Control, Automation and Systems*, pp. 264-269, October 2002

[2] Jin Woo Park, Jae Han Park, Hwa Ra Hur, Jang Myung Lee, Kiyoharu Tagawa, and Hiromassa Haneda, "Capturing a Moving Object using an Active Camera mounted on a Mobile Robot," *Proc. of the 5th Int'l Symp. on Artificial Life and Robotics*, pp. 609-612, Jan. 26-28, 2000.

[3] C.S. Lin, P.R. Chang, and J.Y.S. Luh, "Formulation and optimization of cubic polynomial trajectories for industrial robot," *IEEE Trans. Automatic Cont.*, vol.23, pp. 1066-1074, 1983.

[4] P. K. Allen, A. Tmcenko, B. Yoshimi, and P. Michelman, "Trajectory filtering and prediction for automated tracking and grasping of a moving object," *IEEE International Conference on Robotics and Automation*, pp. 1850-1856, 1992.

[5] Yi Ma, J. Kosecka, and S. S. Sastry, "Vision guided navigation for a nonholonomic mobile robot," *IEEE Transaction on Robotics and Automation*, vol. 15, no.3, pp. 521-536, June 1999.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans., ASME, J. Basic Eng*, Series 82D, pp. 35-45, Mar. 1960.

[7] H. W. Sorenson, "Kalman filtering techniques," *Advances in Control Systems Theory and Applications*, vol. 3, pp. 219-292, 1966.

[8] K. V. Ramachandra, "A Kalman tracking filter for estimating position, velocity and acceleration from noisy measurements of a 3-D radar," *Electro. Technology*, Vol. 33, pp.66-76, 1989.

[9] Yi-Yuan Chen and Kuu-young Young, "An intelligent radar predictor for high-speed moving-target tracking," *TENCON '02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering*, Vol. 3, pp.1638 -1641, 2002.

[10] J. M. Roberts, D. J. Mills, D. Charnley, and C. J. Harris, "Improved Kalman filter initialization using neurofuzzy estimation," *Int'l. Conf. on Artificial Neural Networks*, pp. 329-334, 1995.

[11] J. P. DeCruyenaere and H. M. Hafez, A comparison between Kalman filters and recurrent neural networks," *Neural Networks, 1992. IJCNN., Int'l Joint Conf.*, Vol. 4 , June 1992.

[12] Sorensen. O, "Neural networks for non-linear control," *Control Applications, 1994, Proceedings of the Third IEEE Conference on*, 24-26 Aug. 1994.

[13] P. Partsinevelos, A. Stefanidis, P. Agouris, "Automated spatiotemporal scaling for video generalization," *Proc. of 2001 Int'l. Conf. on Image Processing*, Vol.1, Oct. 2001.

[14] Rafael G. Gonzalez, Richard E. Woods, "Digital Image Processing," Addison-Wesley, pp 47-51, 1993

[15] Russell F. Berg, "Estimation and prediction for maneuvering target trajectories," *IEEE Trans. on Automation Control*, vol. AC-38, Mar. 1983.

[16] Steven M. Lavalle and Rajeer Sharma, "On motion planning in changing partially predictable environments," *The Int'l. Jour. of Robotics Research*, Vol.16, no. 6, pp. 705-805, 1997.

[17] J. T. Feddema and C. S. G. Lee, "Active image feature prediction and control for visual tracking with a hand-eye coordinated camera," *IEEE Trans. Sys. Man Cyber.*, Vol. 20, pp. 1172-1183, 1990.

[18] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator control with image-based visual servo," *IEEE Int'l. Conf. On Robotics and Automation*, pp. 2267-2272, 1991.

[19] T. Kohonen, "Self-Organized formation of topologically correct feature maps," *Biological Cybernetics*, 1982, pp.59-60.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Cooperative Behavior Learning in Distributed Autonomous Robotics System

Chul-Min Hwang, Ji-Yoon Kim, Dong-Wook Lee, and Kwee-Bo Sim

School of Electrical and Electronic Engineering, Chung-Ang University
221, Heuseok-Dong, Dongjak-Gu, Seoul 156-756, Korea

saramsa2@wm.cau.ac.kr, kbsim@cau.ac.kr

Abstract

This paper proposes a learning method of cooperative behavior in distributed autonomous robotic systems. In this paper, the strategies of a robot system are composed of a global strategy and a local strategy. The global strategy controls the whole robots in the system and the local strategy controls some robots in the local area. The global strategy, which is aggregation and dispersion, is decided by artificial immune network. The artificial immune network is one of biological immune network model. The biological immune network copes with thousands invasion from the outside. The local strategy controls some robots to take a task using distributed genetic algorithm.

Keywords: distributed autonomous robotics system, global strategy, local strategy, distributed genetic algorithms, artificial immune network

1. Introduction

Distributed Autonomous Robotic System (DARS) is that each robot perceives its environments such as an object and the other robot's strategy etc., and they determine their strategy independently, and cooperate with the other robots in order to perform the given tasks very well. DARS has no function to integrate information of environment. But a robot individually understands surrounding and strategy of other robots, and decides its strategy autonomously to cooperate with other robots. The whole system is composed of individual robot's strategy. Biological immune system is also a distributed autonomous system. It is not a centralized control system. But it can cope with varying environment autonomously. So immune system is applied by a distributed autonomous robotic system [1,2].

Artificial immune system is the model of a biological immune system. The basic cooperative unions of artificial immune system are B-cell and T-cell, which are two major types of lymphocytes. B-cell takes part in humoral responses that secrete antibodies and T-cells take part in cellular responses that stimulate or suppress cells connected to the immune system. Operation of B-cell and T-cell can express artificial immune network equations [2]. These equations regenerate a stimulated value of antibody. Also antibody is stimulated or suppressed from antigen, other antibodies and T-cell.

In this paper we compose DARS based on artificial immune network and add learning system in DARS. The learning system makes the adaptation for unknown tasks. This learning system is based on distributed genetic algorithms. Therefore proposed DARS has two algorithms. The one is artificial immune network and the other is distributed genetic algorithm so that we present artificial immune system and distributed genetic algorithms. As well we simulate and evaluate proposed system.

2. Immune System

2.1 Clonal Selection

Each lymphocyte (whether B cell or T cell) is genetically programmed to be capable of recognizing essentially only one particular antigen. The immune system as a whole can specifically recognize many thousands of antigens. When an antigen binds to the few cells that can recognize it, they are induced to proliferate rapidly. Within a few days there are a sufficient number to mount an adequate immune response. In other words, the antigen selects for and generates the specific clones of its own antigen- binding cells, which is called a clonal selection. This operates for both B cells and T cells [2].

Lymphocytes that have been stimulated by binding to their specific antigen take the first steps towards cell division. They express new receptors that allow them to respond to cytokines from other cells, which is signal proliferation. The lymphocyte may also start to secrete cytokines themselves. They will usually go through a number of cycles of division, before differentiating into mature cells, again under the influence of cytokines [2].

Fig. 1 shows operation of B cell clonal selection that is basic response of immune system.

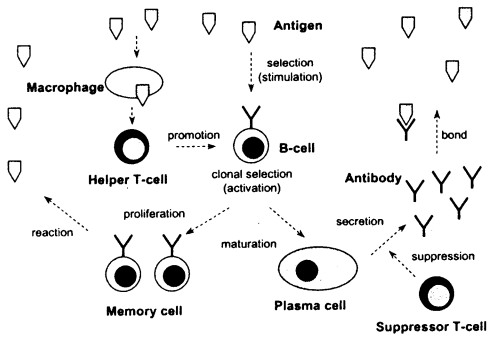


Fig. 1. Operation of T-cell and B-cell clonal selection.

2.2 Idiotopic Network

Tolerance to self-antigens is established during onto chromosome. However, during the neonatal period the unique binding regions of antigen-specific receptors on B-cells (antibody) and T-cells are present at levels that are too low to generate tolerance. This antigen-specific site is called an idiotope. Thus antibodies (B-cells) are stimulated not by antigens but also by other antibodies (B-cells). Jerne who is an immunologist proposed idiotopic network hypothesis (immune network hypothesis) based on mutual stimulus and suppression between antibodies. According to this hypothesis, antibodies interact with each other through idiotope and paratope. Such a network relationship plays an important role that keeps up the concentration of antibody in the immune system. Therefore, immune system is parallel-distributed system that operates on not a unit level but system level [2].

2.4 Immune Network Model Of B-cell and T-cell

Equation (1)-(3) are the modified immune network equations that are modeled relationship of antigen, B-cell (antibody), and T-cell of immune system.

$$S_i(t) = S_i(t-1) + \left((\alpha \frac{\sum_{j=1}^{N}(m_{ij}s_i(t))}{N} - \alpha \frac{\sum_{j=1}^{N}(m_{ki}s_k(t))}{N} + \beta g_i - c_i(t-1) - k_i \right) s_i(t) \quad (1)$$

$$s_i(t) = \frac{1}{1+\exp(0.5 - S_i(t))} \quad (2)$$

$$c_i(t) = \eta(1 - g_i(t))S_i(t) \quad (3)$$

Where $i, j = 0, N\text{-}1$, N is a number of antibody types, $S_i(t)$ is stimulus value of antibody i, $s_i(t)$ is concentration of antibody i, $s_j(t)$ is not concentration of self-antibody but concentration of other robot's antibody getting by communication, $c_i(t)$ is concentration of T-cell which control concentration of antibody, m_{ij} is mutual stimulus coefficient of antibody i and j, g_i is affinity of antibody i and antigen, α,β,η are constants [2].

3. 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 [3]. The second is that each agent has one chromosome, and evolutes mutually by communication in distributed system [4]. The third is that chromosome is divided as part, and each part of chromosome evolutes respectively, finally they are put together [5-7]. 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 and each robot's fitness is evaluated by given task performance.

Fig. 2 shows how the distributed genetic algorithm to be applied to the system. Each robot senses environment and communicates each other. In this case sensing and communication range is limited. Also each robot has one chromosome, and then phenotype corresponds to this chromosome, determines local strategy. The fitness of the chromosome is evaluated from robot's local strategy. The robots exchange the information of the chromosome when they communicate.

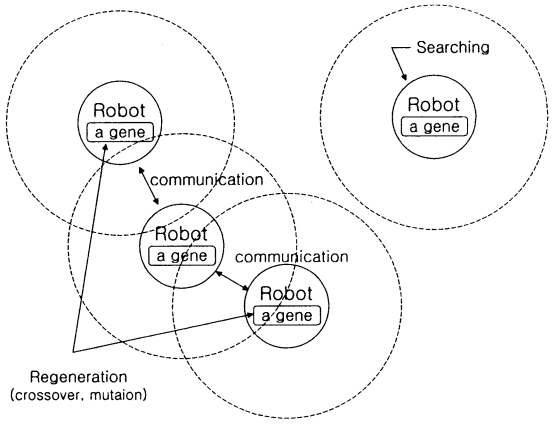


Fig. 2. The application of distributed genetic algorithm in the system

In this time, if the fitness of one agent is lower than the other, the chromosome is regenerated by genetic algorithm. Two robots' chromosomes are selected to the parents. Crossover and mutation generate two children. A child chromosome is selected by roulette wheel selection. The selected chromosome is robot's regenerated chromosome (Fig. 2).

When some robots compose a team, this team has

several chromosomes. Therefore we assign the first robot, which detects an object, for the master robot. The master robot recruits the three robots, which are assigned the slave and obey the master robot's order. Evaluating the team's performance for accomplishing task is equivalent to evaluating master robot's chromosome. Therefore the evaluation value of the team becomes the fitness of master robot's chromosome

A robot searches the task or the order from the master robot. If a robot finds an object that was not detected by the other robots and become the master robot, then the master robot orders some robots by the phenotype of chromosome. The robots in the master robot's communication range are compared with the type of the master robot's chromosome. If their type is equal, we assign the searching robot for slave and order the slave to move to the object. When the master robot orders all slave robots to take a task. However the robot gives up the role of the master robot and searching the other tasks when the master robot fails the recruiting slaves for the defined time. Then the slaves can become the master robot. The change of master robot is equivalent to change applying chromosome. This process makes the system various by changing applied chromosome periodically.

4. Simulation

The simulator proceeds that the robots carry the objects to the target area. The robot's strategies are divided into two parts, which are the global strategy and the local strategy. The global strategy is divided into the aggregation and dispersion. The selection system in the searching is the artificial immune network that selects the strategies, aggregation and dispersion. The carrying is learning the cooperative union of robots' type to increase the performance of the task. The task is carrying the objects to the target area (Fig. 3).

In the artificial immune network, the antigens are determined by how long objects are detected and how many robots are detected. The former relates with aggregation, and the later relates with dispersion.

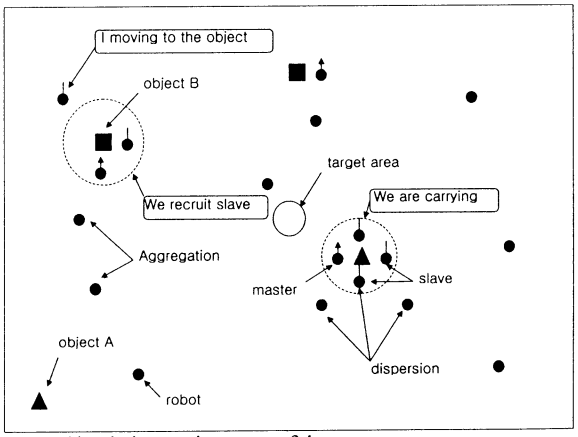


Fig. 3. Simulation environment of the system.

This system needs four robots to carry an object, and the object has proper cooperative union of robots' type. When this cooperative union of robot's type takes the task the speed of carrying is fastest. Also the other cooperative unions have some penalties for the speed of carry. Table 1 shows the efficiency when the robot accomplishes other robot's roles.

Table 1. The robot's efficiency when the robot takes a task that is optimized other robot type.

	R0	R1	R2	R3
R0	1	0.3	0.1	0.3
R1	0.3	1	0.3	0.1
R2	0.1	0.3	1	0.3
R3	0.3	0.1	0.3	1

The each robot has one chromosome for the one type of objects. The chromosome is composed by 6bits binary string and 2 bits binary string expresses one of robot's types. Therefore, the chromosome expresses three of robot's types. These three types and self-robot's type compose a team, which carries an object. Then the chromosome doesn't care that the robots' types repeat.

The chromosome of the robots is evaluated when the robot is assigned to the master robot, composes the team, and the team processes the task. The fitness (f) of the chromosome evaluate by carrying time (t) and distance (d). The equation (1) show how calculate the fitness.

$$f = \frac{t}{d} \tag{1}$$

New chromosomes are generated from evaluated chromosomes when two robots, which are evaluate from the accomplishing the task, contacts in the communication range. The chromosome that has lower fitness between them is regenerated by genetic algorithm. The two chromosomes precede two-point crossover and mutation. Mutation rate is 0.05.

5. Results

Each robot has communication and detection range, of which radius is fifty and. Robots are divided four types and each type has twenty-five robots. The number of total robots is a hundred. The objects, which robot must carry to target area, are divided two types (type A and type B). The target area is center of simulation field. If an object arrived the target area, the object is regenerated randomly in the simulation field. Each object has best combination of robot's type and table 2 shows the combination of robot's type. The crossover rate is 1 and mutation rate is 0.1 when the genetic algorithm operates between evaluated two robots.

Table 2. The combination of robot's type for each object.

Object	Robot's type			
A	R0	R1	R2	R3
B	R0	R2	R2	R3

Fig. 4 is average fitness landscape for evaluated robots that accomplish the task to carry the object. In the Fig 2, x-axis displays time and y-axis displays fitness value. The fitness of the best efficiency is 1. The average fitness for object A and object B converge in the Fig. 4. The average fitness of task A converges into 1 gradually. Also the average fitness of task B converges into 0.9. What The average fitness is 1 indicate that all evaluated chromosomes' fitness are 1. Therefore the almost chromosomes for task A are best chromosome. However the average fitness of task B is low, because of R2's fitness. The best fitness of R2 is not 1 because R2's chromosome for task B always includes R2 that is oneself.

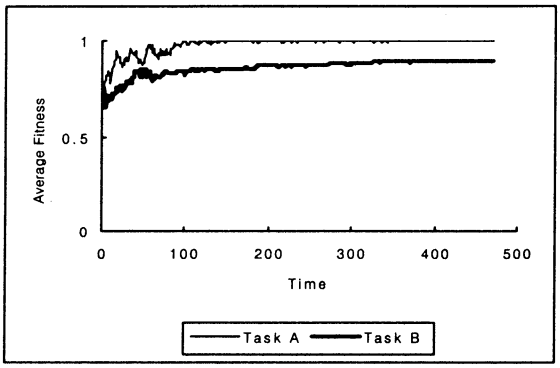


Fig. 4. Average fitness of evaluated robots.

Fig. 5 shows the number of evaluated chromosomes for each task. In the Fig. 5, x-axis displays time and y-axis displays number of robot. We recognize that the number of evaluated chromosomes increase gradually.

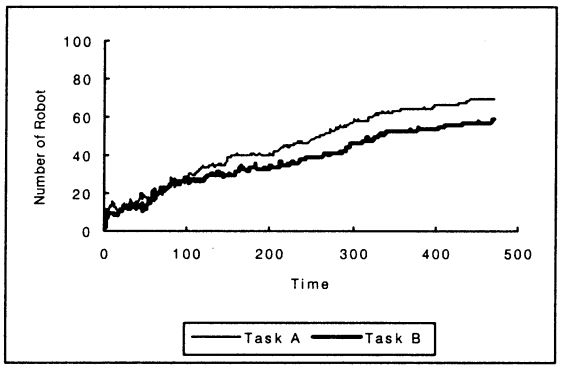


Fig. 5. The number of evaluated chromosomes for each task

6. Conclusion

In this paper we propose cooperative behavior learning based on distributed genetic algorithm. Also we evaluate our proposed system. The system learns for the cooperative union of chromosome's type to accomplish the task that moves an object. Also the searching based on artificial immune network increases speed of learning. Each robot has only one chromosome, but distributed genetic algorithm finds best cooperative union and spread the other robots.

Acknowledgements

This research was supported by the Braintech 21 from Ministry of Science and Technology of Korea.

References

[1] D.W. Lee, and K.B Sim, "Artificial Immune Network-based Cooperative Control in Collective Autonomous Mobile Robots," *Proceedings of the 6th IEEE International Workshop on ROBOT AND HUMAN COMMUNICATION(RO-MAN)*, pp. 58-63, 1997. 9. 29 – 10. 1

[2] D.W. Lee, H.B. Jun, and K.B. Sim, "Artificial Immune System for Realization of Cooperative Strategies and Group Behavior in Collective Autonomous Mobile Robots", *Proceedings of The 4th International Symposium on Artificial Life and Robotics*, vol 1, pp. 232-235, 1999. 1. 20

[3] A. Loraschi et. al., "Distributed Genetic Algorithms with An Application to Portfolio Selection Problems," *Proc. of IC Artificial Neural Nets and Genetic Algorithms*, pp. 384-387, 1995.

[4] E. Horiuchi, K. Tani, "Behavior Learning of Group of Mobile Robots with a Distributable Genetic Algorithm," *J of RSJ(Japan)*, vol. 11, no. 8, pp. 1212- 1219, 1993.

[5] D.W. Lee, K.B. Sim, "Behavior Learning and Evolution of Collective Autonomous Mobile Robots using Distributed Genetic Algorithms," *2nd ASian Control Conference*, vol.2, pp. 675-678, 1997. 7. 24(Seoul, Korea)

[6] H.B. Jun, K.B. Sim, "Behavior Learning and Evolution of Collective Autonomous Mobile Robots based on Reinforcement Learning and Distributed Genetic Algorithms," *Proceedings of the 6th IEEE International Workshop on ROBOT AND HUMAN COMMUNICATION (RO-MAN)*, pp. 248-253, 1997. 9.29 - 10.1(Sendai, Japan)

[7] D.W. Lee, K.B. Sim, "Cooperative Behavior of Collective Autonomous Mobile Robots Based on Online Learning and Evolution," *Proceedings of 7th International Conference on Neural Information Processing(ICONIP 2000)* , 2000. 11. 14-18.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Fingerprint Matching Algorithm Based on the Artificial Immune System

Jae-Won Jeong, Tae-Hoon Kim, Dong-Wook Lee, and Kwee-Bo Sim

School of Electrical and Electronic Engineering, Chung-Ang University
221, Heuseok-Dong, Dongjak-Gu, Seoul 156-756, Korea

asellus@wm.cau.ac.kr, kbsim@cau.ac.kr

Abstract

Fingerprints have been widely used in the biometric authentication because of its performance, uniqueness and universality. In this paper, we keep an eye on the self-nonself discrimination in the biological immune system for the higher fingerprint identification performance. We propose a fast and reliable fingerprint matching algorithm based on the generation of the cytotoxic T-cell for the recognition of the MHC protein in order to discriminate between self and nonself. Our matching algorithm makes a global pattern matching that means a self-space of the template fingerprint image and MHC detector set from the test fingerprint image in the first stage based on the minutiae and the directional field of the fingerprint, then the algorithm organizes a topological local structure of the minutiae in the second stage for the robust matching against translation and rotation. In order to reduce matching time for a fast matching, algorithm applies second local structure matching to the MHC detectors that already recognized in the global pattern matching stage.

Keywords : fingerprint, artificial immune system, self-recognition model

1. Introduction

Biometrics is a technology for personal identification by the extraction of his (or her) physiological or behavioral characteristics. By a excellent security and convenience based on the property of using his (or her) physiological or behavioral characteristics, biometrics are highlighted as new personal identification method that substitutes for the existing personal identification method such as a key, ID card, password, etc [1].

In his (or her) biological information, fingerprint is a collection of a flow that consists of ridges and valleys. Fingerprint has the property of permanence and uniqueness. And fingerprint has higher valuation than other biological informations as hand vein, iris, and face for the index that composed of uniqueness, permanence, collectability, and performance [1][2]. So the fingerprint identification technology is used in a various field of security application. Many researches are accomplished for the fingerprint identification, and now fingerprint matching and identification is achieved based on the minutiae of fingerprint. Fingerprint matching algorithm based on the topological structure of minutiae has good performance in the identification in spite of noisy fingerprint images. However, these matching methods can't get a fast matching time by many numbers of matching times [3].

In this paper, we propose a fast and reliable fingerprint matching algorithm that derives inspiration from the process of the generation of the cytotoxic T-cell for the generation of the MHC protein in order to discriminate between self and nonself [4]. Also our matching algorithm considers the topological structures of minutiae for the robust fingerprint matching against translation and rotation.

2. Immune System

2.1 Biological Immune System

Biological Immune System is the 2nd defensive system of the creatures that defenses the self against foreign invaders such as a bacterium, a virus. Also the creatures develop highly advanced immune systems through the process of the evolution. Such biological immune system is a distributed autonomous system without the central organ that controls a function of immune system. Also biological immune system has abilities to manage, learn, and memorize the information about the antigen, and has a ability to eliminate the 'non-self' through the 'self' - 'non-self' discrimination [5]. From the viewpoint of the information processing, immune system functions as a highly advanced parallel-processing intelligent system [6]. So many researches about the modeling of biological immune systems for industrial application have been doing actively. Artificial immune system is the artificial modeling of biological immune system.

2.2 MHC recognition for self-recognition

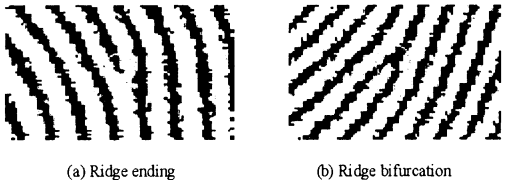
A function of biological immune system is accomplished by the operation of B-cell and T-cell. B-cell that is generated in the bone marrow creates antibodies to eliminate antigens. T-cell is generated in

the thymus and is classified into three classes [7]; helper T-cell, cytotoxic T-cell, and suppressor T-cell. Helper T-cell activates B-cell to accelerate the secretion of antibodys. Cytotoxic T-cell recognizes infected cells by antigens and kills infected cells. Also Suppressor T-cell suppresses the activity of the immune system after the immune system is activated by antibodies.

Cytotoxic T-cell has two recognition parts to tell whether a cell is a self cell or not and to examine whether an antigen exists in the self or not. The MHC recognition part recognizes the MHC protein that tells whether a cell is self or not, and antigen recognition part recognizes the foreign invaders as an antigen. In this paper, the modeling of the MHC recognition part is made based on the minutiae and the directional field of the fingerprint image, and is applied to the fingerprint matching.

3. Minutiae extraction for fingerprint identification

The minutiae based on the ridge structure of the fingerprint is used for the fingerprint identification. There are two types in the minutiae; ridge ending and ridge bifurcation. Fig. 1 shows examples of the ridge ending and ridge bifurcation.



(a) Ridge ending (b) Ridge bifurcation

Fig. 1. Ridge ending and ridge bifurcation.

In general, fingerprint images can have noises by wet or dry finger, humidity in the air, dirts on the fingerprint sensor or the surface of the finger. Also these facts make the analysis of fingerprint image and the extraction of minutiae difficult [2]. So specialized image processing technology for the fingerprint image processing is needed to eliminate noises of fingerprint image and to show the prominent structure of the ridge. Fig. 2 shows the procedure of the minutiae extraction in brief.

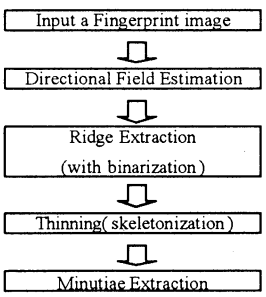


Fig. 2. Process of the minutiae extraction.

4. Fingerprint Matching Algorithm based on the Artificial Immune System

Fingerprint matching process is very important to the fingerprint identification technology. In this paper, we propose the robust fingerprint matching algorithm based on the artificial immune system for more fast and robust matching. The existing fingerprint matching methods based on the minutiae of the fingerprint uses all of the minutiae in the template fingerprint images and in the input fingerprint image for the fingerprint matching. In contrast with the existing methods, proposed matching algorithm in this paper do 1st stage matching by the use of the self-space that is constructed by the distribution of the minutiae and the directional field of the template fingerprint image and the MHC recognition part that is constructed from the self-space of the input fingerprint image. Then 2nd stage matching is accomplished by the minutiae within the matched area in the 1st stage matching to decide whether they are matched or not. Also these processing can accelerate the matching speed because of the number of the minutiae that is used in the matching stage is reduced. In the 2nd stage matching, the local structures for the minutiae that is matched in the 1st stage matching are made to do robust matching against rotation, translation, and other transformations.

First, the algorithm divides the input fingerprint image into the block of size 16× 16 pixels. Each block is expressed by the binary numbers and six bits are allocated to each block. Left two-bits represents whether the minutiae is exist in the block or not. Also, right four-bits represents the direction of the block by the 4-bit expression of the directional field value in the block. The directional field describes the normalized 8-directions and the 4-bit representation is used for the sequential representation of the 8-directions. Repeat this process for all of the block in the fingerprint image, then the set of the binary strings are made to organize the self-space. Fig. 3 illustrates the construction procedure of the self-space of the collection of the binary strings that each string has a length of 6× (number of blocks in the row) bits.

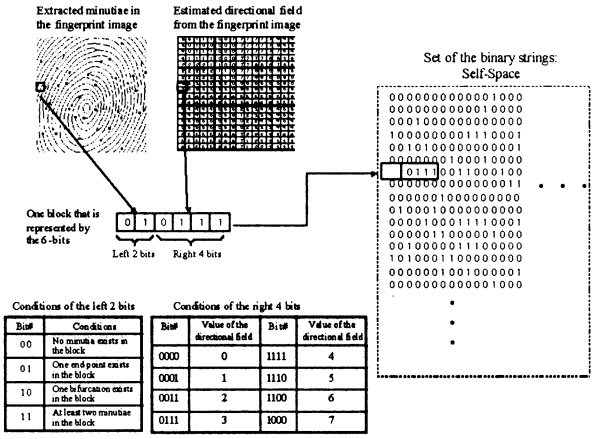


Fig. 3. construction of the self-space from the self-space From the 6× (number of blocks in the row)bit strings.

that form the self-space, l bits are extracted from the center of the binary strings of the self-space, then the extracted bit strings organize the MHC detector set that is composed of the N MHC detector strings of l bits long. Fig. 4 shows the generation of the MHC detector set.

Fig. 4. Generation of the MHC detector set from the self-space.

MHC detector set that is generated from the input fingerprint image is matched to the binary string set of the self-space that is generated from the template fingerprint image (the size of the self-space of the template fingerprint image is the same as the size of the self-space of the input fingerprint image). MHC detector string in the MHC detector set that is made in the input fingerprint image is matched to the binary string of the self-space of the template fingerprint image by shifting the one bit toward the self-space strings of the template fingerprint image to find the position of the best matching score.

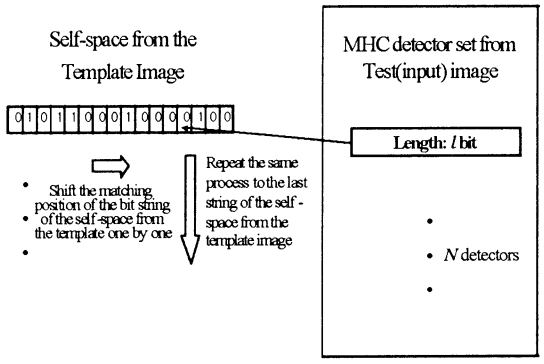


Fig. 5. MHC detector set from the input fingerprint image is matched to the self-space of the template.

However, by the characteristics of the bit-pattern matching and the matching process in Fig.5, the possibility that the MHC detector from the input fingerprint image is matched in the 'unexpected' position of the self-space from the template fingerprint image cannot be neglected. So the additional process that is showed in the next is applied for the correction of this problem.

#1. When the process in Fig. 5 is finished, calculate the *position difference* of the best scored matching position of the self-space of the template image and the MHC detector of the input image. The total number of the *position difference* is equal to the total number of the MHC detector in the MHC detector string set.

$$PositionDifference[N] = i - N$$

N : row number of the matched MHC detector

i : row number of the matched self-space

#2. Use the *position differences* that are calculated in #1 to evaluate the *average value* of the position differences.

$$AveragePositionDifference = \frac{\sum_{i=0}^{N-1} PositionDifference[i]}{N}$$

N : total number of the MHC detector string

#3. Using #1 and #2, calculate the following equation for each MHC detector:

$$diff1 = PositionDifference[N](\text{old}) - AveragePositionDifference$$

N : row number of the MHC detector

(1) If $|diff1| > \pm B$, B is the boundary value that is decided by empirical results.

Do the matching process in Fig.5 again for that MHC detector. The range of the matching in this process islimited by the *lower bound* and the *upper bound*.

$$lower\ bound = N + AveragePositionDifference - B$$

$$upper\ bound = N + AveragePositionDifference + B$$

N : row number of the MHC detector

After this limited-range matching process is accomplished, calculate

$$diff2 = PositionDifference[N](\text{new}) - AveragePositionDifference$$

If $|diff2| > \pm B$, the matching score of that MHC detector is set to zero. Else the corrected matching score of that MHC detector remains unchanged and comes to the next matching process.

(2) If $|diff1| \leq \pm B$,

No further operation is allowed. Also matched score of that MHC detector comes to the next matching process.

In the process to match the self-space from the template with the MHC detector set from the input fingerprint image, for the matched position of the MHC detector and self-space with the best score, the algorithm generates the local structure that is composed of the minutiae in the input and template fingerprint images. Also the local structure considers the minutiae in the matched blocks as a center point. Then the proposed algorithm does the 2nd stage matching toward the constructed local structures. The parameters of the local structure in the 2nd stage are invariant against rotation and translation[8][9]. So using the local structure in the 2nd stage matching guarantees the robust fingerprint matching against translation, rotation, and other transformations of the fingerprint image. When the matching score exceeds the limit score, That MHC detector string is recognized as 'self' This process is repeatedly applied to the MHC detector set that consists of the N MHC detectors, the result of the matching process is used to make a decision of whether the input

fingerprint image is matched to the template fingerprint image or not. The structure of the local structure and the self-recognition by the 2nd stage matching are showed in Fig.6 briefly.

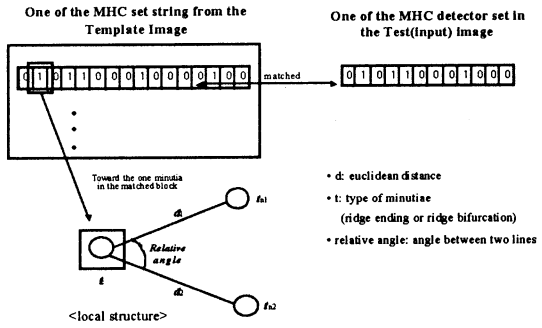


Fig. 6. self-recognition by the local structure matching when the MHC detector string is matched.

5. Experimental Results

In our experiments, we used fingerprint image of size 288(wid) × 320(hig) pixels. To apply the propose algorithm in this paper, fingerprint image is divided into the block of size 16 × 16 pixels, and the fingerprint image has total 18(row) × 20(column) blocks. Then the most outer blocks in the fingerprint image are excluded from the process, and self-space is made of 18 binary strings that each binary string has a length of 96 bits(16 blocks). 10 MHC detector strings are made from the self-space of the input fingerprint image, and each MHC detector has a length of 72 bits. Experiments are accomplished about the 1st stage matching by the self-space of the template image and the MHC detector set of the input image. 2nd stage matching based on the local structures of the minutiae is left to future works.

Total of 18 fingerprint images that are collected from 6 individuals are used in the experiments. By the experimental results of the 1st stage matching process, all of the fingerprint images that are used in the experiments are properly matched to the right individuals.

6. Conclusions

In this paper, we proposed the fingerprint matching algorithm based on the self-recognition model by the MHC recognition part of the cytotoxic T-cell from the artificial immune system. First stage matching is obtained by the use of the self-space of the template and the MHC detector set of the input fingerprint image. Also this process reduces the candidate set of the minutiae and the error rate of the matching by the false minutiae due to noisy images. MHC detector string that is matched to the binary string of the self-space of the template is used to construct the local structure which regards the minutia in the matched MHC detector as the center minutia. The parameters of the local structure are invariant against rotation and translation of the fingerprint image, and these parameters of the local structure is used to the 2nd stage matching for an improvement of the reliability in the matching process. By the experiments, we showed that the fingerprint matching algorithm based on the self-recognition model is useful for the fingerprint identification. Future works are the realization of the 2nd stage matching based on the local structure of the minutiae, the correction of problems that are found from the experimental results. Also experiments in more large fingerprint database are important tasks for our research.

Acknowledgment

This research was supported by the industry-university-research consortium program from the Seoul city and Small & Medium Business Administration (SMBA) of Korea

References

[1] K. B. Sim, C. B. Ban, J. Y. Sim, "Development of Intelligent Fingerprint Recognition System," *The Journal of KASBIR*, vol. 1, no. 2, pp. 111-119, 2001.

[2] A. K. Jain, Lin H., S. Pankanti, R. Bolle, "An Identify-Authentication System Using Fingerprints," *Proc. of the IEEE*, vol. 85, pp. 1365-1388, 1997.

[3] A. Wahab, S. H. Chin, E. C. Tan, "Novel Approach to automated fingerprint recognition," Proc. of IEEE Conf. on Vision, *Image and Signal Processing*, Vol. 145, pp. 160-166, 1998.

[4] K. B. Sim, D. W. Lee, "Change Detection Algorithm based on Positive and Negative Selection of Developing T-cell," *Journal of Fuzzy logic and Intelligent Systems*, vol. 13, No. 1, pp. 119-124, 2003.

[5] J. W. Yang, D. W. Lee, K. B. Sim, Y. S. Choi, D. I. Seo, "Intrusion Detection Algorithm based on Artificial Immune System," *Proc. on ICCAS2002*, pp. 110-114, 2002.

[6] D. Dasgupta, *Artificial immune systems and their application*, Springer-Verlag Berlin Heidelberg, 1999.

[7] T. Tomio, *The meaning of the immune system*, Han-Wool, 1998.

[8] X. Jiang, W. Y. Yau, "Fingerprint Minutiae Matching Based on the Local And Global Structires," *IEEE Proc. on Pattern Recognition*, vol. 2, pp. 1038-1041, 2000.

[9] D. P Mital, E. K. Teoh, "An Automated Matching Technique for Fingerprint Identification," *Proc. on KES '97.*, vol. 1, pp. 142-147, 1997.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Optimal Design of the 2-Layer Fuzzy Controller with Vision System

Kwang-Sub Byun, Kwang-Seung Heo, Chang-Hyun Park, and Kwee-Bo Sim

School of Electrical and Electronic Engineering, Chung-Ang University
221, Heuseok-Dong, Dongjak-Gu, Seoul 156-756, Korea

ks4070@wm.cau.ac.kr, kbsim@cau.ac.kr

Abstract

The robot with complex and various functions is required. A robot with various inputs generates various outputs. A proper controller is needed to control such a complex robot. In this paper, we introduce a novel algorithm, 2-layer fuzzy controller. It can control robustly and has the small number of rules in many input variables. It can not only deal with various inputs but also generate various outputs. Also, we propose an advanced color detecting system using a specialized color histogram. This vision system is used in robust target tracking. The main problem in the fuzzy system is how to design the fuzzy knowledge base. For this specific problem, we optimize the fuzzy controller using the schema co-evolutionary algorithm to find a global optimal solution. Through the design of the optimal fuzzy controller using the schema co-evolutionary algorithm and the simulation of the mobile robot, we verify the efficacy of the 2-layer fuzzy controller and the schema co-evolutionary algorithm together.

Keywords: 2-layer fuzzy controller, schema co-evolutionary algorithm, color detecting vision system

1. Introduction

Nowadays, the robot with complex and various functions is required. The robot with various inputs generates various outputs. The function of such a robot is still simple, but it will be various. Distance, optical and verbal information and even electronic map using GPS are used to sense the surroundings. Therefore, the existing algorithm cannot perform perfectly the desired task to control the versatile intelligent robot.

Many algorithms for the behavior of a robot have proposed. Among them, the most used method is the fuzzy control. However, the total numbers of fuzzy rules and adjustable system parameters increase exponentially with the number of input variables in standard fuzzy reasoning process [1]. It imposes a heavy burden on the system in view of the control speed and cost. So, the system with many input variables needs the fuzzy reasoning system that can control robustly and has the low number of rules. The 2-layer fuzzy controller was proposed to solve these problems [2].

By the way, the main problem in the fuzzy controller is how to design the fuzzy knowledge base. There are many approaches to design the optimal fuzzy controller. In particular, evolutionary computation has received large attention in the past years. Although SGA provides many opportunities to obtain a global optimal solution, the performance of SGA is more or less limited depending on the predefined fitness function [3]. For the solution to this problem, we optimize the fuzzy controller using the Schema Co-Evolutionary Algorithm (SCEA) to find a global optimal solution [4].

The field of robot vision has the most various application among sensors of a robot. The field of robot vision is roughly classified into three applications; edge detection of an object for the obstacle avoidance, path following, target tracking. The most important application is the target tracking [5]. Most of the target tracking are used in robot soccer system. However, it is specialized the certain application. In this paper, we propose an advanced color detection algorithm using the specialized color histogram.

Therefore, in this paper we optimize the 2-Layer Fuzzy Controller (2LFC) using the Schema Co-Evolutionary Algorithm (SCEA). In this application, the 2LFC has a color detecting system for mobile target tracking. So, a robot can follow another robot. Section 2 describes the structure of the 2LFC and compares it with the multilevel fuzzy systems. In Section 3 we introduce the SCEA, and we describe the advanced color detection system and apply it to the 2LFC in Section 4. The experiment on the color detection and the SCEA for design of the 2LFC are performed in Section 5. Finally, Section 6 presents concluding remarks.

2. 2-Layer Fuzzy Controller

2.1 Structure of the 2LFC

Chung and Duan classified the hierarchical fuzzy systems into three multistage structures, namely, incremental, aggregated, and cascaded [6]. For the aggregated one, inputs are only allowed to pass to the first reasoning stage consisting of a number of independent sub-stages. The outputs from the first stage form the inputs to the successive stage and such an arrangement can be extended for more stages.

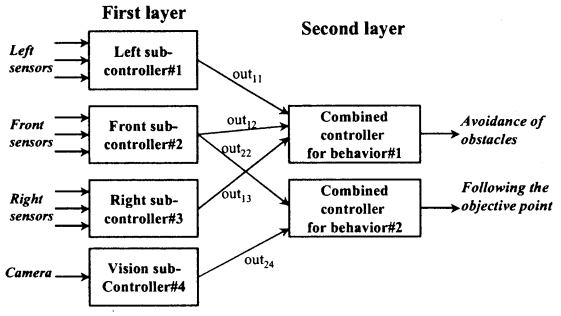


Fig. 1. The structure of the 2-Layer Fuzzy Controller

Fig. 1 shows the structure of the 2LFC. This figure shows the structure, which has 4 sub-controllers in the first layer and 2 combined controllers in the second layer. The process of the 2LFC is depicted in detail in [2].

As shown in this figure, it is similar to the aggregated structure of multilevel fuzzy relational system. Also, it is similar to the shape of the neural network. However, the 2LFC has three differences. First, 2LFC has a specialized structure from the aggregated structure. The 2LFC limits the fuzzy reasoning level to the 2-level, that is, 2-layer. In case of multilevel fuzzy system, a solution to increasing stage is not reliable and it cannot secure the stability of the entire system. The 2LFC, however, can secure the stability of the system by limiting the fuzzy reasoning level to the 2-level. Second, 2LFC has more than two modules in the final level to that final output is connected. Nowadays, objectives to be controlled are very complex and various. So, they have not only many various inputs but also many various outputs. Multilevel fuzzy system can deal with many various inputs but not many various outputs. Because 2LFC has more than two modules in the final layer, it can deal with many various inputs and many various outputs. Finally, it is likely to layered structure of the artificial neural network. 2LFC, however, has a fuzzy controller in each module; it is known as the node in the artificial neural network. Also, 2LFC has not weights. In this case, 2LFC has nothing to do with the artificial neural network.

2.2 Application of 2-layer fuzzy controller

The robot has 3 sensors in front, 3 in left, and 3 in right. Total 9 ultrasonic sensors are used for measurement of the distance. These sensors have same type, but the number of them is too many to control. So, we divide them into 3 input groups. In Fig. 1, inputs of the sub-controller are 3 sensors in left, 3 in front, and 3 in right. Vision camera has a different type from ultrasonic sensors. So, the input of the sub-controller#4 is vision camera. 'out_{ij}' represents the output of j^{th} sub-controller and the input of i^{th} combined controller.

We apply the 2LFC to the following of the target point. So, combined controller#2 generates this behavior and needs the position of a target. It needs information of the perception of the target, that is, the direction and the distance of a target, and inputs of the combined controller#2 are out_{22} and out_{24}. Front sub-controller generates the distance from the robot to the target and vision sub-controller generates the direction of the target. Each module has a different fuzzy reasoning system. In this application, simplified method of Sugeno is used as the fuzzy inference engine [7]. The reason is that output of each controller is a constant value.

3. Optimal Design Using the SCEA

3.1 Schema co-evolutionary algorithm

In order to find optimal fuzzy controller, in this paper, we use the Schema Co-Evolutionary Algorithm (SCEA). The SCEA was proposed in [4].

The SGA has four major steps for one generation, which are evaluation, selection, crossover, and mutation. The SCEA, however, has an additional step, parasitizing, before the selection. After all strings in the host-population are evaluated, some of them are selected randomly for each schema of the parasite-population and then parasitized by the corresponding schema. It evaluates the strings newly generated from parasitizing and then measure the fitness improvement between the original string and the parasitized one. It replaces the parasitized string having the largest improvement value with the corresponding string for each schema. Using the amount of the improvement it can assign the fitness of the each schema in the parasite-population. The fitness of an individual in the parasite-population is calculated by parasitizing process. So, parasite-population takes four steps that is parasitizing, selection, crossover, and mutation. The parasitizing process and SCEA process are explained in detail in [4].

3.2 Design of 2-layer fuzzy controller

We apply the SCEA to the optimal design of 2LFC with vision system that is described in section 2.2. Each fuzzy rule in each fuzzy controller has the following form:

if $X1$ is $A1$ and $X2$ is $A2$ and $X3$ is $A3$ then Y is c

where, $X = (X1, X2, X3)$ and Y are linguistic variables and $(A1, A2, A3)$ and c are their respective linguistic values. Especially $(A1, A2, A3)$ are forms of general membership functions and c is a fuzzy singleton in simplified method. The fuzzy rule with three membership functions needs eight points: $m1 \sim m8$. If these points are determined, three membership functions of a trapezoidal form are determined as Fig. 2. Also constants in consequent needs 27 points: $c1 \sim c27$, because each fuzzy controller with 3 inputs and 3 membership functions has 27 rules. Each fuzzy rule needs 35 parameters. One parameter needs 6 bits in a string. So, a fuzzy rule needs total 210 bits that mean the length of a string.

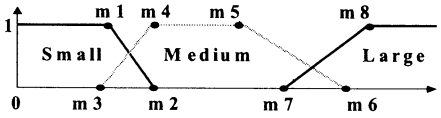


Fig. 2. Membership functions to be designed

In the 2LFC, combined controller is more important than sub-controllers. First of all, the SCEA designs the combined controller with 3 inputs. And then, after adding a left sub-controller with 3 inputs, the SCEA designs this sub-controller. Next, after adding a right sub-controller with 3 inputs, it is designed. Finally, after adding a front sub-controller with 3 inputs, it is designed by the SCEA. In case of the fuzzy rule for target tracking, vision system is added to the system while the SCEA is performed.

The evaluation part in the SCEA process is changed according to the applied problem. The fitness function in this application has the following expression.

$$f = (pos/11)\times(1-near/50)\times(15/time) \quad (1)$$

The distance of a robot moving is divided into 11 sectors (*pos0*~*pos11*). The *pos0* represents a start point and the *pos11* does an aimed point. The closer a robot moves to the wall, the lower the fitness is given. The *near* represents the number of closeness to the wall within 50cm and the maximum number of closing is *50*. The time represents the elapsed time that the robot moves from the *pos0* to the *pos11* and *15* is the minimum elapsed time. Consequently, the highest fitness value is given to the robot that moves most rapidly to the goal point and has no collision and no closeness within 50cm to obstacles.

The robot in front has the 2LFC by this process except vision system. In order to design the 2LFC for the robot that follows the robot, vision system is included and a certain distance between two robots is maintained during a fitness evaluation.

4. Advanced Color Detection

In order to raise the ability of the autonomous mobile robot, this paper implements the color detecting system.

We implement the vision system that can simply perceive a certain color region with a web camera. This vision system is similar to the way used in the robot soccer system [8]. However, a specialized color histogram is added. Fig. 3 shows the process of the advanced color detection using a specialized color histogram.

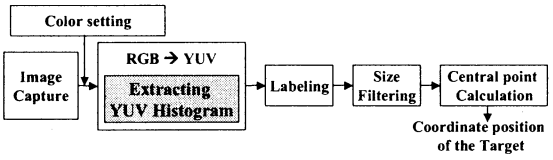


Fig. 3. Advanced color detecting algorithm

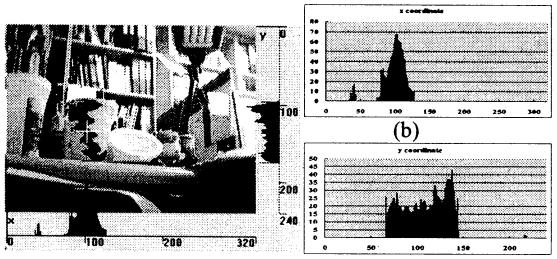


Fig. 4. Example of a specialized color histogram
(a) Picture of an experiment (b) x histogram (c) y histogram

After capturing the image each frame, the system extracts 2 histograms —horizontal and vertical axis histogram—, while it exchanges RGB image to YUV image. And then, through labeling, size filtering, and calculation of the central point with each 1-dimensional histogram, it finds a coordinate position of the target. Each histogram in this application has the numbers of a desired YUV value in each position. Fig. 4 shows the example of the proposed histogram.

The advantages of this system are the fast speed and robustness. For the image of 320×240 pixel, vision system without the specialized histogram deals with 2-dimensional image each step of the vision process. It takes 230,720 cycles. On the other hand, vision system with the specialized color histogram deals with two 1-dimenstional images. It takes 78,480 cycles. Therefore, this system is three as fast as one without extracting histogram. Also, this system needs not a filtering process in its output. Robustness of the system is experimented in the next section.

5. Experiments and Results

5.1 Comparison SCEA with SGA

We compared with the SGA to verify the efficacy of the SCEA. For simple comparison, we designed only one fuzzy controller, that is, combined controller. The population size of SGA is set for 100. The host and parasite-population sizes of the SCEA are set for 40 and 30 respectively. The sampling size n is set for 2. Therefore, the total evaluation number per each generation is 40+30×2, which is the same number of the population size of SGA. We replicated 100 runs. Also elite-preserving strategy is used. Each search was run to 150 generations in Table 1. *Pc* and *Pm* represent the crossover probability and mutation probability of population in SGA and host-population in SCEA, respectively. *Pc* and *Pm* of the parasite-population have fixed 0.05 and 0.1, respectively. The best result represents that the fitness is over 0.95.

These results show that the SCEA can find more than SGA does in the same condition. The algorithm with the high *Pm* found more. Especially, the SCEA with the 0.01 *Pm* found almost all the best result in 100 runs.

Table. 1. The number of best results and average generation of best results in 100 runs

P_c	P_m	Num. of best results		Avg. generation of best results	
		SGA	SCEA	SGA	SCEA
0.5	0.001	52	87	23.60	32.82
	0.01	89	99	26.88	24.93
	0.1	92	96	46.74	47.49
0.7	0.001	53	93	31.23	31.17
	0.01	91	100	19.82	23.31
	0.1	82	100	57.74	49.27

5.2 Optimal design of 2LFC

As mentioned in section 3.2, the SCEA for each fuzzy controller was performed. Every process of the SCEA achieved best fitness value 1.0. The membership functions that are obtained for each fuzzy controller are depicted in Fig. 5.

Step 1 found the individual with the best fitness in the 127th generation, step 2 did it in the 36th one, step 3 did it in the 5th one, and step 4 did it in the 2nd one. Also, we can see that only one robot navigation takes 4547msec and two robots navigation, which means a robot following, takes 5922msec. Naturally, a robot following takes longer than one robot navigation.

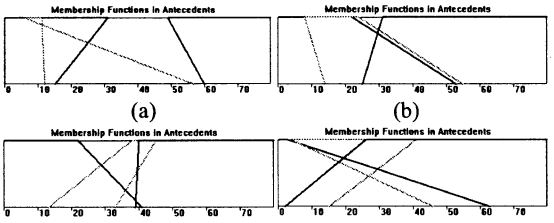


Fig. 5. Membership functions after SCEA process
(a) Step 1: combined controller (b) step 2: left sub-controller
(c) step 3: right sub-controller (d) step 4: front sub-controller

5.3 Color detection

For two vision systems, we experimented on target tracking. One vision system has the extraction process of the specialized color histogram. Another vision system has not. Fig. 6 shows the horizontal coordinate changes according to target movement in time.

As shown in this figure, the advanced color detecting system with the specialized color histogram has a robust target tracking and needs not a filter. But, general color detecting system needs a certain filter that can remove its uncertainty.

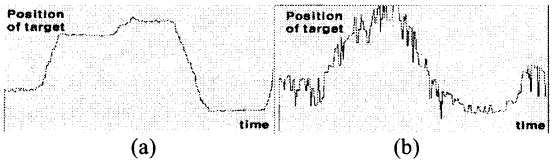


Fig. 6. Horizontal coordinates of a target in time
(a) Vision system with the histogram (b) the system without it

6. Conclusions

In this paper, we design the 2LFC with the advanced vision system using the SCEA. The vision system using specialized color histogram is used in robust target tracking. The SCEA provides more various searching space with small population size than SGA. It has chance to converge more rapidly than SGA. So, the SCEA can find more rapidly the global optima than SGA. The 2LFC with 9 ultrasonic sensors and 1 vision system was used for the robot following. Each controller was designed in turns by the SCEA. Consequently, the robust 2LFC with vision system was designed.

The vision system in this paper can't recognize what is the target. This problem can be solved by addition of the edge detection and other algorithm. Also, it is expected that the optimal design of the 2LFC will be widely used.

Acknowledgment

This research was supported by the Brain Science and Engineering Research program sponsored by Korea Ministry of Science and Technology.

References

[1] F.L. Chung and J.C. Duan, "On multistage fuzzy neural network modeling," *IEEE Trans. on Fuzzy Systems*, vol. 8, no. 2, pp. 125-142, Apr. 2000.

[2] K.B. Sim, K.S. Byun, and C.H. Park, "2-Layer fuzzy controller for behavior control of mobile robot," *Jour. of Korea Fuzzy Logic and Intelligent Systems Society*, vol. 13, no. 3, pp. 287-292, June 2003.

[3] J.R. Koza, "Genetic evolution and co-evolution of computer programs," *Artificial Life II*, Addison-Wesley, vol. 10, pp. 603-629, 1991.

[4] K.B. Sim, H.B. Chun, D.W. Lee, "Dynamic behavior control of autonomous mobile robots using schema co-evolutionary algorithm," *Proc. of IEEE International Symposium on Industrial Electronics (ISIE 2001)*, vol. 1, pp. 560 -565, 12-16 June 2001.

[5] T.M. Chen, R.C. Luo, "Mobile target tracking using hierarchical grey-fuzzy motion decision-making method," *Proc. on IEEE International Conference on Robotics and Automation (ICRA '00)*, vol.3, pp. 2118 -2123, 24-28 Apr. 2000.

[6] J.C. Duan and F.L. Chung, "Multilevel fuzzy relational systems: structure and identification," *Soft Computing 6*, Springer-Verlag 2002, pp. 71-86, 2002.

[7] H. Kang and K.B. Sim, *Intelligent Information System*, DaeyoungSa, 2001.

[8] J.H. Kim, et al., *Robot soccer engineering*, KAIST press, pp. 109-172, 2002.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Discretization of Takagi-Sugeno Fuzzy Observer-Based Output-Feedback Control Using Intelligent Digital Redesign

H. J. Lee and J. B. Park
Dept. of Electrical. and Electronic. Eng.,
Yonsei University
Seoul 120-749, Korea

Y. H. Joo
School of Electronic. and Information. Eng.,
Kunsan National University
Kunsan 573-701, Korea

Abstract

An intelligent digital redesign technique (IDR) for the observer-based output feedback Takagi-Sugeno fuzzy control system with fuzzy outputs is developed. The IDR condition is cubically parameterized as convex minimization problems of the norm distances between linear operators to be matched. The global exponential stability condition is embedded and the separation principle is shown. The proposed method can be utilized as an efficient design tool for a reliable nonlinear sampled-data output-feedback control.

1 Introduction

Dynamical behaviors of most physical systems are characterized by a set of differential equations in analog setting. From a standpoint of control engineering, it is no wonder that the researches have been focused on the development of various design techniques for the analog control. However, digital implementation of a control is desirable especially when the designed control utilizes some sophisticated algorithms that require considerable amount of computation efforts.

Yet another efficient approach is so-called digital redesign (DR), to convert the well-designed analog control into the equivalent digital one maintaining the properties of the original analog control system in the sense of state-matching, by which the benefits of both the analog control and the advanced digital technology can be achieved. It has been noted that the digital redesign schemes basically work only for a class of linear systems [1]. For that reason, it has been highly demanded to develop some intelligent digital redesign (IDR) methodology for complex nonlinear systems, in which the first attempt was made by Joo [4]. They synergistically merged both the Takagi-Sugeno (T-S) fuzzy-model-based control and the digital redesign technique for a class of nonlinear systems. Chang extended the intelligent digital redesign to uncertain T-S fuzzy systems [3] and elaborated it [2]. However, until now, no tractable method for IDR tackling on the observer-based output-feedback T-S fuzzy system with fuzzy outputs has been proposed.

Motivated by the above observations, this paper proposes an effecient design methodology of an observer-based output-feedback digital control for T-S fuzzy systems via IDR. The IDR problem is viewed as convex minimization problems of the norm distances between linear operators to be matched. The stability condition is easily embedded and the separations principle is explicitly shown.

2 T-S Fuzzy Systems

Consider a T-S fuzzy system in which the $\boldsymbol{I/O}$ form is formulated as follows:

$$\dot{x}_c(t) = \sum_{i=1}^{q} \theta_i(z(t))(A_i x_c(t) + B_i u_c(t))$$
$$y_c(t) = \sum_{i=1}^{q} \theta_i(z(t)) C_i x_c(t)$$

where $x_c(t) \in \mathbb{R}^n$, $u_c(t) \in \mathbb{R}^m$, and $y_c(t) \in \mathbb{R}^p$. The subscript 'c' means the analog control, while the subscript 'd' will denotes the digital control in the sequel. Throughout this paper, a well-constructed analog observer-based fuzzy-model-based control is assumed to be pre-designed, which will be used in redesigning the digital control, in the $\boldsymbol{I/O}$ forms:

$$\dot{\widehat{x}}_c(t) = \sum_{i=1}^{q} \theta_i(z(t))(A_i \widehat{x}_c(t) + B_i u_c(t) + L_c^i(y_c(t) - \widehat{y}_c(t))) \quad (1)$$
$$\widehat{y}_c(t) = \sum_{i=1}^{q} \theta_i(z(t)) C_i \widehat{x}_c(t)$$

and

$$u_c(t) = \sum_{i=1}^{q} \theta_i(z(t)) K_c^i \widehat{x}_c(t) \tag{2}$$

Let the estimation error $e_c(t) = x_c(t) - \widehat{x}_c(t)$, then we obtain the augmented system is

$$\dot{\chi}_c(t) = \sum_{i=1}^{q}\sum_{j=1}^{q} \theta_i(z(t))\theta_j(z(t)) \times \begin{bmatrix} A_i + B_i K_c^j & -B_i K_c^j \\ 0 & A_i - L_c^i C_j \end{bmatrix} \chi_c(t) \tag{3}$$

where $\chi_c(t) = \left[x_c(t)^T, e_c(t)^T\right]^T$.

3 Discretization of T-S Fuzzy Systems

This section discusses the discretization of the hybrid T-S fuzzy system. Consider the hybrid T-S fuzzy system

$$\dot{x}_d(t) = \sum_{i=1}^{q} \theta_i(z(t))(A_i x_d(t) + B_i u_d(t)) \tag{4}$$

and $u_d(t) = u_d(kT)$ is the piecewise-constant control input vector to be determined, in the time interval $[kT, kT+T)$, and $T > 0$ is a sampling period. There are several methods for discretizing a linear time-invariant (LTI) analog system. Unfortunately, these discretization methods cannot be directly applied to the discretization of the analog T-S fuzzy system since the defuzzifed output of the T-S fuzzy system is not LTI but implicitly time-varying [5]. Moreover, it is further desired to maintain the polytopic structure of the discretized T-S fuzzy system for the construction of the digital fuzzy-model-based control. Thus we need a mathematical foundation for the discretization of the analog T-S fuzzy system.

Assumption 1 *Assume that the firing strength of the ith rule, $\theta_i(z(t))$ is approximated by their values at time kT, that is, $\theta_i(z(t)) \approx \theta_i(z(kT))$ for $t \in [kT, kT+T)$. Consequently, the nonlinear matrices $\sum_{i=1}^{q} \theta(z(t))A_i$ and $\sum_{i=1}^{q} \theta(z(t))B_i$ can be approximated as constant matrices $\sum_{i=1}^{q} \theta(z(kT))A_i$ and $\sum_{i=1}^{q} \theta(z(kT))B_i$, respectively, over any interval $[kT, kT+T)$.*

Theorem 1 *The pointwise dynamical behavior of* (4) *can be efficiently approximated by*

$$x_d(kT+T) \approx \sum_{i=1}^{q} \theta(z(kT))(G_i x_d(kT) + H_i u_d(kT)) \tag{5}$$

Proof: See the reference [5]. ■

To reconstruct $x_d(t)$ at $t = kT, k \in \mathbb{Z}^+$, the following ***I/O*** form of a discrete fuzzy-model-based observer is adopted

$$\begin{aligned} \widehat{x}_d(kT+T) &= \sum_{i=1}^{q} \theta_i(z(kT))(G_i \widehat{x}_d(kT) + H_i u_d(t) \\ &\quad + L_d^i(y_d(kT) - \widehat{y}_d(kT))) \\ \widehat{y}_d(kT) &= \sum_{i=1}^{q} \theta_i(z(kT)) C_i \widehat{x}_d(kT) \end{aligned} \tag{6}$$

For the digital control of (4), we take a control in the following ***I/O*** form:

$$u_d(t) = \sum_{i=1}^{q} \theta_i(z(kT))(K_d^i \widehat{x}_d(kT) + F_d^i y_d(kT)) \tag{7}$$

for $t \in [kT, kT+T)$, where K_d^i and F_d^i is the digital control gain matrices to be redesigned for the ith rule.

Let the error be $e_d(kT) = x_d(kT) - \widehat{x}_d(kT)$. Then the closed-loop system with (5), (6) and (7) is constructed to yield

$$\begin{aligned} &\chi_d(kT+T) \\ &\approx \sum_{i=1}^{q}\sum_{j=1}^{q} \theta_i(z(kT))\theta_j(z(kT))\theta_k(z(kT)) \\ &\quad \times \begin{bmatrix} G_i + H_i K_d^j + H_i F_d^j C_h & -H_i K_d^j \\ 0 & G_i - L_d^i C_j \end{bmatrix} \chi_d(kT) \end{aligned} \tag{8}$$

where $\chi_d(kT)^T = \left[x_d(kT)^T, e_d(kT)^T\right]^T$.

Corollary 1 *The pointwise dynamical behavior of* (3) *can also be approximately discretized as*

$$\chi_c(kT+T) \approx \sum_{i=1}^{q}\sum_{j=1}^{q} \theta_i(z(kT))\theta_j(z(kT))\Phi_{ij}\chi_c(kT) \tag{9}$$

where

$$\begin{aligned} \Phi_{ij} &= \exp\left(\begin{bmatrix} A_i + B_i K_c^j & -B_i K_c^j \\ 0 & A_i - L_c^i C_j \end{bmatrix} T\right) \\ &= \begin{bmatrix} \phi_{ij}^{11} & \phi_{ij}^{12} \\ \phi_{ij}^{21} & \phi_{ij}^{22} \end{bmatrix}, \quad \forall (i,j) \in \mathcal{I}_Q \times \mathcal{I}_Q, \end{aligned}$$

and $\chi_c(kT) = \left[x_c(kT)^T, e_c(kT)^T\right]^T$,

$\phi_{ij}^{11} = \exp\left((A_i + B_i K_c^j)T\right)$, $\phi_{ij}^{22} = \exp\left((A_i - L_c^i C_j)T\right)$,

$$\phi_{ij}^{12} = -\int_0^T e^{(A_i + B_i K_c^j)(T-\tau)} B_i K_c^j e^{(A_i - L_c^i C_j)\tau} d\tau,$$

and $\phi_{ij}^{21} = [0]_{n\times n}$

Proof: It is straightforwardly proven by Theorem 1. ■

4 Intelligent Digital Redesign

The IDR problem can be formulated as follows:

Problem 1 *Given the well-designed gain matrices K_c^i and L_c^i for (2) and (1), find K_d^i and F_d^i for (7), and L_d^i for (6) such that the followings are satisfied:*

(i) *The state $\chi_d(kT)$ of (8) matches the state $\chi_c(kT)$ of (9) at every sampling time instance $t = kT$, as closely as possible.*

(ii) *The digitally controlled system is globally exponentially stable (GES).*

Theorem 2 *Suppose (8) is GES in the sense of Lyapunov; then the zero equilibrium points $x_d(t) = [0]_{n\times 1}$ and $e_d(kT) = [0]_{n\times 1}$ of the hybrid control system that consists of (4), (6), and (7) are also GES.*

Proof: The proof is omitted due to lack of space. ■

Consider the first objective in Problem 1. In parallel with (8) and (9), to realize $\chi_c(kT+T) = \chi_d(kT+T)$ under the assumption $\chi_c(kT) = \chi_d(kT)$, it is desired to determine K_d^i, F_d^i, and L_d^i in such a way that the following matrix equality conditions

$$\phi_{ij}^{11} = G_i + H_i K_d^j + H_i F_d^j C_h \tag{10}$$

$$\phi_{ij}^{12} = -H_i K_d^j \tag{11}$$

$$\phi_{ij}^{22} = G_i - L_d^i C_j \tag{12}$$

holds for all triplet $(i, j, h) \in \mathcal{I}_Q \times \mathcal{I}_Q \times \mathcal{I}_Q$.

Remark 1 *Equalities (10) and (11) may be solved for K_d^i and F_d^j if $n \leqslant p$ and $n \leqslant m$, and H_i and C_i both are nonsingular. Similarly, (12) may be solved for L_d^i provided that $n \leqslant p$ and C_i is nonsingular. Such conditions are rarely fulfilled even in LTI case. In addition, it should be addressed that (10), (11), and (12) are hardly solved in many case of T-S fuzzy-model-based control, since each variable K_d^i, F_d^i, and L_d^i should satisfy r^3, r^2, and r different equality conditions, respectively. In addition, guaranteeing the GES of (8) weights the problem down.*

In order for the difficulties to be resolved, an alternative approach is applied by relaxing Problem 1 and searching K_d^i, F_d^i, and L_d^i such a way that the norm distances between ϕ_{ij}^{11} and $G_i + H_i K_d^j + H_i F_d^j C_h$, ϕ_{ij}^{12} and $-H_i K_d^j$, and ϕ_{ij}^{22} and $G_i - L_d^i C_j$, respectively, are minimized, hence, Problem 1 is restated as follows:

Problem 2 *Given the well-designed gain matrices K_c^i and L_c^i for (2) and (1), find K_d^i and F_d^i for (7), and L_d^i for (6) such that the followings are satisfied:*

1) *Minimize γ_1 and γ_2 over K_d, F_d, and L_d subject to $\left\|\phi_{ij}^{11} - G_i - H_i K_d^j - H_i F_d^j C_h\right\| < \gamma_1$, $\left\|\phi_{ij}^{12} + H_i K_d^j\right\| < \gamma_1$, and $\left\|\phi_{ij}^{22} - G_i + L_d^i C_j\right\| < \gamma_2$ in the sense of the spectral norm measure.*

2) *The system (8) is GES in the sense of Lyapunov.*

Remark 2 *System (9) can be viewed as a convex combination of quadratically parameterized sub-closed-loop systems. Similarly (8) is also a convex combination of cubically parameterized sub-closed-loop systems. It yields the minimization problem of the norm distances between r^2 of ϕ_{ij}^{11} and r^3 of $G_i + H_i K_d^j + H_i F_d^j C_h$. It is necessary to examine whether the upper bound γ_1 is guaranteed or not when $\theta_h(z(kT)) \neq 1, h \in \mathcal{I}_Q$.*

Theorem 3 *Suppose $\left\|\phi_{ij}^{11} - G_i - H_i F_d^j C_h\right\| < \gamma_1$, $\forall (i, j, h) \in \mathcal{I}_Q \times \mathcal{I}_Q \times \mathcal{I}_Q$; then $\left\|\phi_{ij}^{11} - G_i - H_i F_d^j \sum_{h=1}^{q} \theta(z(kT)) C_h\right\| < \gamma_1$ holds.*

Proof: From the fact that the induced 2-norm is a convex operator and $\mathbf{Co}\{C_1, \ldots, C_r\}$ is obviously a convex set, the claim directly follows. ■

Theorem 4 *If there exist symmetric positive matrices P_1, Q_2, symmetric matrices X_{ij}, Y_{ij}, and matrices K_d^i, F_d^i, N_i with compatible dimensions, and possibly small scalars $\gamma_1 > 0, \gamma_2 > 0$ such that the following two generalized eigenvalue problems (GEVPs) have solutions*

GEVP 1: $\underset{P_1, X_{ij}, K_d^i, F_d^i}{\text{Minimize}}$ γ_1 *subject to*

$$\begin{bmatrix} -\gamma_1 & (\bullet)^T \\ \phi_{ij}^{11} - G_i - H_i K_d^j - H_i F_d^j C_h & -\gamma_1 I \end{bmatrix} \prec 0 \tag{13}$$

$$\begin{bmatrix} -\gamma_1 & (\bullet)^T \\ \phi_{ij}^{12} + H_i K_d^j & -\gamma_1 I \end{bmatrix} \prec 0, \quad (i, j, h) \in \mathcal{I}_Q \times \mathcal{I}_Q \times \mathcal{I}_Q \tag{14}$$

$$\begin{bmatrix} -P_1 + X_{ij} & (\bullet)^T \\ \left(\frac{G_i + H_i K_d^j + H_i F_d^j C_h + G_j + H_j K_d^i + H_j F_d^i C_h}{2}\right) & -P_1 \end{bmatrix} \prec 0 \tag{15}$$

$$[X_{ij}]_{r\times r} = [X_{ij}]_{r\times r}^T \succ 0, \quad (i, j) \in \mathcal{I}_J \times \mathcal{I}_Q \tag{16}$$

$$\begin{bmatrix} -P_1 & (\bullet)^T \\ I & -P_1 \end{bmatrix} \preccurlyeq 0 \tag{17}$$

GEVP 2: $\underset{Q_2, Y_{ij}, N_i}{\text{Minimize}} \quad \gamma_2 \quad$ *subject to*

$$\begin{bmatrix} -\gamma_2 Q_2 & (\bullet)^T \\ (\phi_{ij}^{22})^T Q_2 - G_i^T Q_2 + C_i^T N_j^T & -\gamma_2 I \end{bmatrix} \prec 0 \tag{18}$$

$$\begin{bmatrix} -Q_2 + Y_{ij} & (\bullet)^T \\ \left(\left(\frac{G_i^T Q_2 - C_i^T N_j^T}{2} \right) + \left(\frac{G_j^T Q_2 - C_j^T N_i^T}{2} \right) \right) & -Q_2 \end{bmatrix} \prec 0, \quad (i,j) \in \mathcal{I}_Q \times \mathcal{I}_Q \tag{19}$$

$$\left[Y_{ij} \right]_{r \times r} = \left[Y_{ij} \right]_{r \times r}^T \succ 0, \quad (i,j) \in \mathcal{I}_J \times \mathcal{I}_Q \tag{20}$$

then, $\chi_d(kT)$ *of* (5) *via the redesigned digital fuzzy-mode-based control* (7) *closely matches* $x_c(kT)$ *of the discretized version of* (9). *Furthermore,* (5) *is GES in the sense of Lyapunov stability criterion, where* $N_i = Q_2 L_d^i$, $\mathcal{I}_J = \{1, 2, \ldots, j\}$, *and* $(\bullet)^T$ *denotes the transposed element in symmetric positions.*

Proof: First, the first constraint in the first objective of Problem 2 holds if and only if the following inequality are satisfied:

$$(\phi_{ij}^{11} - G_i - H_i K_d^j - H_i F_d^j)^T (\phi_{ij}^{11} - G_i - H_i K_d^j - H_i F_d^j C_h) \prec \gamma_1^2 I$$

which is equivalent to (13). We can again establish a similar argument to that above with the second constraint of Problem 2, to obtain LMI. Next consider the last constraint in the first objective of Problem 2. Introducing a free matrix variable W with a full column rank of n, we have

$$\left\| (\phi_{ij}^{22} - G_i + L_d^i C_j)^T \right\|_2 \leqslant \gamma_2 = \widehat{\gamma}_2 \left\| W \right\|_2 \tag{21}$$

where $\gamma_2 = \gamma_2 / \left\| W \right\|_2$ is a positive scalar. Choosing W such that $P_2 \succ W^T W$, and from the definition of the induced 2-norm, (21) holds if LMI (18) holds, where $\widehat{\gamma}_2$ is abused as γ_2. The remaining LMIs (15), (16), (17), (19), and (20) follow from the standard Lyapunov exponential stability theorem in [6], which completes the proof of the theorem. ■

Remark 3 (separation principle) *It is important to address that, since the searching variables* $P_1, X_{ij}, K_d^i, F_d^i$ *for the digital control in GEVP 1 and* Q_2, Y_{ij}, N_i *for the discrete observer in GEVP 2 of Theorem 4 are not coupled, the IDR for* (7) *and* (6) *can be performed independently, which indicates that the separation principle holds for IDR of the observer-based output-feedback fuzz-model-based control.*

5 Conclusions

In this paper, a new IDR has been proposed for the observer-based output-feedback fuzzy-model-based control. The developed technique formulated the given IDR problem as constrained convex optimization problems so that the powerful and flexible numerical algorithms can be utilized. The flexibility of the LMIs enables one to incorporate the stability of the redesigned system into the IDR algorithm. The separation principle was clearly shown.

Acknowledgements

This work was supported by KOSEF R01-2001-000-00316.

References

[1] W. Chang , J. B. Park , H. J. Lee, and Y. H. Joo, "LMI approach to digital redesign of linear time-invariant systems," *Proc. Inst. Elect. Eng., vol. 149, pt. D,* no. 4, pp. 297-302, July, 2002.

[2] W. Chang, J. B. Park, and Y. H. Joo, "GA-based intelligent digital redesign of fuzzy-model-based controllers," *IEEE Trans. Fuzzy Systems,* vol. 11, no. 1, pp. 1-10, 2003.

[3] W. Chang, J. B. Park, Y. H. Joo, and G. Chen, "Design of sampled-data fuzzy-model-based control systems by using intelligent digital redesign," *IEEE Trans. Circuits Syst. I,* vol. 49, no. 4, pp. 509-517, 2002.

[4] Y. H. Joo, L. S. Shieh, and G. Chen, "Hybrid state-space fuzzy model-based controller with dual-rate sampling for digital control of chaotic systems," *IEEE Trans. Fuzzy Syst.,* vol. 7, no. 4, Aug., 1999.

[5] Z. Li, J. B. Park, and Y. H. Joo "Chaotifying contiuous-time T-S fuzzy systems via discretization," *IEEE Trans. Circuits Syst. I,* vol. 48, no. 10, pp. 1237-1243, 2001.

[6] E. Kim and H. Lee, "New approaches to relaxed quadratic stability conditions of fuzzy control systems," *IEEE Trans. Fuzzy Syst.,* vol. 8, no. 5, pp. 523-534.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

DNA Computing for an Absolute 1-Center Problem: An Evolutionary Approach

Zuwairie Ibrahim[1], Osamu Ono[2], Marzuki Khalid[3], and Yusei Tsuboi[4]

[1,2,4]Institute of Applied DNA Computing, Meiji University
1-1-1 Higashimita, Tama-ku, Kawasaki, Kanagawa, JAPAN

[3]Centre for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia
Jalan Semarak, 54100 Kuala Lumpur, MALAYSIA

(zuwairie, ono, tsuboi)@isc.meiji.ac.jp, marzuki@utmkl.utm.my

Abstract

In this paper, the possibility of DNA-based computing to solve an absolute 1-center problem by molecular manipulations is presented. The DNA-based computation is designed in such a way that every path is encoded by oligonucleotides and the paths' length is directly proportional to the length of oligonucleotides. Then, gel electrophoresis is performed in order to separate the respective DNA molecules according to length. In this work, the tedious affinity-purification operation is not involved and could be eliminated from the procedure completely. One expectation arise from this paper is that it is possible to verify the instance absolute 1-center problem using DNA computing by laboratory experiments.

Keywords: DNA computing, length-based computation, absolute 1-center problem

1 Introduction

The facility location optimization has many applications reported to date mostly in operations research and network design problems. It has been used in placement of internet web proxies [1], agglomeration of traffic [2], and web server replications in a content distribution network [3]. One of the facility location optimization is p-center problems. Even if the number p=1 but the network is large, the problems are almost NP-complete. It is then become relevant to develop a new method based on DNA computing where polynomial-time computation may be developed. For these reasons, the authors concentrate on solving an absolute 1-center problem using DNA-based computing.

So far, this problem has been solved extensively by non-molecular or mathematical approaches and performed on sequential silicon based computer as computing platform. Some examples are greedy method [4] and approximation algorithms [5]. Hence, the authors confidently claim that the proposed approach is the first effort to solve the problem by DNA computing.

The absolute 1-center problem, even it involved shortest path computation as part of the problem, it requires computing the longest path among a set of shortest paths. It also requires to compute the shortest path among a set of longest paths. In this approach, the edge weights are handled by the DNA strands and the actual weights are approximated by the directly proportional length DNA strands. Hence, the important information during the computation is DNA length. The computation only considers about the starting node and end node encoded by DNA strands. As a result, the actual nodes encoded within the DNA strands will not be computed.

2 Problem Formulation

The 1-center problem is to locate a facility on the points of a given network or graph so as to minimize the maximum cost of a path between a demand node and its nearest facility. A weighted undirected network or graph $G = (V, E)$ as depicted in Figure 1 is used to model the problem where V and E are the vertices and edges of the graph respectively.

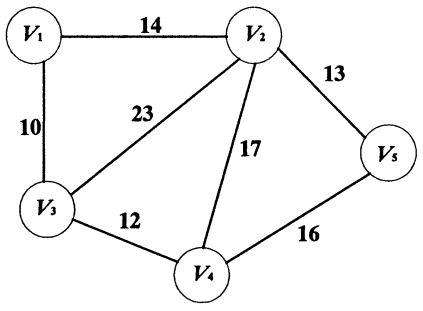


Figure 1. A weighted undirected graph $G = (V, E)$.

Normally, to solve this, one can note the shortest path between a particular node to its nearest node as in Table 1 and choose a node such that the maximum entry in its row in the matrix is smallest among the maximum entries of all rows. The underlined bold numbers highlights the maximum entries for each row. It is clear that the answer is to locate at node V_4 with a maximum distance of 22.

Table 1. Node-To-Node Distance By A Matrix

	V_1	V_2	V_3	V_4	V_5
V_1	0	14	10	22	**27**
V_2	14	0	**23**	17	13
V_3	10	23	0	12	**28**
V_4	**22**	17	12	0	16
V_5	27	13	**28**	16	0

3 Computing with DNA

The overall computation described in this section consists of five steps altogether. Step 1 through step 3 is crucial for the generating initial pool, random route formation, whereas step 4 through step 6 is designed for the computation process and applied to the initial pool of solution.

Step 1: Firstly, the random sequence strands correspond to all nodes (and its complements) and distances (and its complements) are created. Let *O_i (i = 1,....., V)* and $\overline{O}\_i$ *(i = 1,....., V)* be the fixed length random sequences correspond to all nodes in the graph and its complements respectively. In the same way, let *O_d (d = 1,....., E)* and $\overline{O}\_d$ *(d = 1,....., E)* be the variable length random sequences correspond to all the distances and its complements respectively. The length *l* of *O_d* and $\overline{O}\_d$ will be directly proportional to the weight distances. For instance, let say, for a constant proportional increase of 2, all the random sequences are placed in Table 2 and Table 3. For the sake of convenience, in Table 2 and Table 3, in order to avoid clumsy presentation, we will use symbolic representation of the distance sequences. Recall that these sequences would be occurred in any order of nucleotides and its length will be directly proportional to the distances. At the end of this step, oligonucleotides $\overline{O}\_d$ and $\overline{O}\_i$ are synthesized.

Step 2: Let *i* be the start node of a path, *d* the path's distance and *j* the end node of that path. Oligonucleotides representing every path between two nodes in the graph are synthesized as follows:

synthesize oligonucleotides
O_i-d-j_end as **HR** *O_i* + **ALL** *O_d* + **ALL** *O_j*
O_j-d-i_end as **HR** *O_j* + **ALL** *O_d* + **ALL** *O_i*
O_i-d-j as **HR** *O_i* + **ALL** *O_d* + **HL** *O_j*
O_j-d-i as **HR** *O_j* + **ALL** *O_d* + **HL** *O_i*

where '**ALL**' represents the whole DNA strand, '**HL**' the left half part, '**HR**' the right half part and '+' is a join.

Table 2 Fixed Length Node Random Sequences

Nodes	DNA Code *O_i*, (5' → 3')	Complement $\overline{O}\_i$, (3' → 5')
V_1	TATT	ATAA
V_2	GCGG	CGCC
V_3	GTCG	CAGC
V_4	AGAA	TCTT
V_5	TCCG	AGGC

Table 3 Variable Length Distance Random Sequences

Edge Cost	DNA Code *O_d* (5' → 3')	Complement $\overline{O}\_d$ (3' → 5')
D {10}	20	$\overline{20}$
D {12}	24	$\overline{24}$
D {13}	26	$\overline{26}$
D {14}	28	$\overline{28}$
D {16}	32	$\overline{32}$
D {17}	34	$\overline{34}$
D {23}	46	$\overline{46}$

However, because of this combination, the edges strands *O_i-d-j*, *O_i-d-j_end*, *O_j-d-i*, and *O_j-d- i_end* are influenced by the additional nodes bases and hence, unable to handle the weight of each edges. To compensate the additional bases, the strands corresponds to distance sequences are shorten according to the additional bases and the node sequences in the edge strands are left unchanged. All the modified oligonucleotides are listed in Table 4. This modification however, decreases the generality of the approach because the weight of edges less than 3 is not allowed.

Step 3: All modified oligonucleotides followed by all complement oligonucleotides ($\overline{O}\_i$ and $\overline{O}\_d$) are inserted into a test tube and DNA ligase reaction is performed in which random routes through the graph are formed. Then, in order to make sure that no sticky end strands exist in the solution, polymerase enzyme is used for the extension where the sticky end strands are altered to become blunt end strands.

Table 4. Modified Edge Sequences

Edges	DNA Code (5' → 3')
$V_3 - V_1$	CG 14 TATT TT 14 GTCG CG 16 TA TT 16 GT
$V_4 - V_3$	AA 18 GTCG CG 18 AGAA AA 20 GT CG 20 AG
$V_2 - V_5$	GG 20 TCCG CG 20 GCGG GG 22 TC CG 22 GC
$V_1 - V_2$	TT 22 GCGG GG 22 TATT TT 24 GC GG 24 TA
$V_5 - V_4$	CG 26 AGAA AA 26 TCCG CG 28 AG AA 28 TC
$V_4 - V_2$	AA 28 GCGG GG 28 AGAA AA 30 GC GG 30 AG
$V_2 - V_3$	GG 40 GTCG CG 40 GCGG GG 42 GT CG 42 GC

Step 4: Produce another five solutions from the initial solution, N_0.

Step 5: For each solution (or each row);

i. Select a particular node as a starting point and find the shortest path from the starting node to another nodes.
ii. Store the strands representing each shortest path into a solution.
iii. Gather all together the solution representing all shortest paths.
iv. Search the longest strand among the shortest strands, which indicates the maximum path of a particular row in Table 1.

Step 6: Gather all together the solutions that represent all the longest paths of each row. Search the shortest strand. Find the starting node of that path to get the answer of the absolute 1-center problem.

The fourth step could be done by merely divides the solution N_0 equally into five solutions and marks the solutions as N_1, N_2, N_3, N_4, and N_5. Since the dividing operation is a volume decreasing operation, it is important, therefore, to increase the volume of solutions to a sufficient amount. The only operation, which is able to increase the volume of the solutions, is the polymerase chain reaction (PCR) operation. Besides, this operation selects the routes that begin with a particular node for each test tube obtained in the previous step. Five types of primers will be employed where one primer, encoding the 3'-half of starting node is assigned to one test tube as shown in Figure 2.

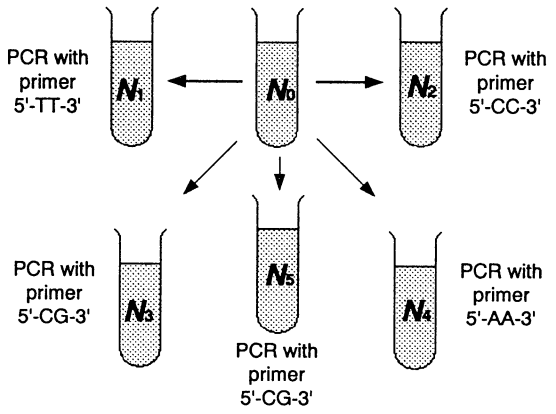


Figure 2. The dividing operation for solution N_0, and the subsequent PCR operation.

Next, each solution obtained from previous operation, N_1, N_2, N_3, N_4, and N_5 are divided equally into another 4 test tube to produce 20 test tubes all together. The DNA strands with a particular end node are amplified and consequently, the volume of solutions is increase as well as depicted in Figure 3.

For better understanding, take a look closely at the molecules of solution N_{12} only. After two PCR operations are accomplished, there should be a numerous number of strands representing the beginning node, V_1 and the end node, V_2. Some of them are shown in Figure 4. Next, gel electrophoresis technique will separates these strands in term of length and the shortest length is chosen in order to obtain the shortest path. As a result, the strands $V_1 \rightarrow V_2$ will be chosen. The output solution is called N_{12}'.

The same procedure will be executed to the solution N_{13}, N_{14}, and N_{15} results in 3 solutions, N_{13}', N_{14}', and N_{15}' consisting the shortest strands $V_1 \rightarrow V_3$, $V_1 \rightarrow V_3 \rightarrow V_4$, and $V_1 \rightarrow V_2 \rightarrow V_5$ respectively. These strands are shown in Figure 5. All these four solutions, N_{12}', N_{13}', N_{14}', and N_{15}' will be gathered together and due to the previous gel electrophoresis outputs, the longest strand will be selected among the shortest strands. Meaning that the strand N_{15}' will be selected. The resultant solution will be called as N_1'. The same procedures for N_1 is repeated to the solution N_2, N_3, N_4, and N_5 and the output solution N_2', N_3', N_4', and N_5' respectively will be produced. At this stage, the sixth step is already completed.

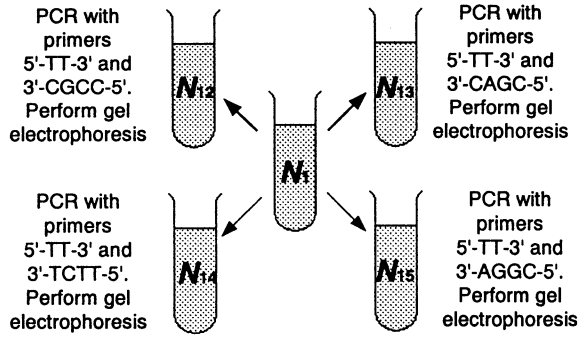


Figure 3. The dividing operation for solution, N_1 and the subsequent PCR operation.

TT	22	GCGG
AA	$\overline{22}$	CGCC

(a)

TT	14	GT	CG	42	GCGG
AA	$\overline{14}$	CA	GC	$\overline{42}$	CGCC

(b)

TT	14	GT	CG	20	AG	AA	30	GCGG
AA	$\overline{14}$	CA	GC	$\overline{20}$	TC	TT	30	CGCC

(c)

Figure 4. (a) Strand $V_1 \rightarrow V_2$ (b) Strand $V_1 \rightarrow V_3 \rightarrow V_2$, and (c) Strand $V_1 \rightarrow V_3 \rightarrow V_4 \rightarrow V_2$.

Lastly, the sixth step begins with collecting the solution N_1', N_2', N_3', N_4', and N_5'. According to the previous gel electrophoresis results, the solution representing the shortest strand will be chosen manually. This solution indicates the actual answer of the absolute 1-center problem. Basically, the proposed algorithm runs in $O(n^2)$ lab operations. If the symmetry property of undirected graph, as considered in this paper is exploited, the time complexity of the proposed approach becomes $O(n\log n)$ where n is the number of vertices in the graph or network.

TT	14	GTCG
AA	$\overline{14}$	CAGC

(a)

TT	14	GT	CG	20	AGAA
AA	$\overline{14}$	CA	GC	$\overline{20}$	TCTT

(b)

TT	22	GC	GG	22	TCCG
AA	$\overline{22}$	CG	CC	$\overline{22}$	AGAA

(c)

Figure 5. (a) Strand $V_1 \rightarrow V_3$ (b) Strand $V_1 \rightarrow V_3 \rightarrow V_4$, and (c) Strand $V_1 \rightarrow V_2 \rightarrow V_5$.

4 Conclusions

This objective of this paper is to certify the feasibility of DNA computing in an absolute 1-center problem. The contents of this paper also imply a possibility to verify the proposed procedures by experimental work. Since the computing platform is now different, it is not relevant to make a comparison between the proposed approach and another existing approaches. Moreover, this computing platform is still infant compared to that matured conventional silicon-based computation. Despite all the limitations, the usefulness and strong potentials of DNA-based computational approach provides a compelling reason to explore a lot of possible applications of DNA computing and solving an absolute 1-center problem is one of them.

References

[1] Li B, Golin M, Italiano G, Deng X, Sohraby K (1999), On the optimal placement of web proxies in the internet. Proceedings of 18th annual joint conference of the IEEE computer and communications societies, 1999, pp. 1282-1290

[2] Andrews M, Zhang L (1998), The access network design problem. Proceedings of the 39th annual IEEE symposium on foundations of computer science, 1998

[3] Jamin S, Jin C, Jin Y, Raz D, Shavitt Y, Zhang L (2000), On the placement of internet instrumentations. Proceedings of 19th annual joint conference of the IEEE computer and communications societies, 2000, pp. 26-30

[4] Gonzalez T (1985), Clustering to minimize the maximum inercluster distance. Theoretical computer science 38: 293-306

[5] Mihelic J, Robic B (2002), Approximation algorithms for k-center problem: an experimental evaluation. Proceedings of operations research, 2002

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

GA based Production Simulator by New Individual Expression, Rough Crossover and Constant Individual Input

Hidehiko Yamamoto* and Etsuo Marui

Faculty of Engineering,
Gifu University
1-1, Yanagido, Gifu-shi, 501-1193, Japan
* yam-h@cc.gifu-u.ac.jp

Abstract: This paper is focused on the GA based production simulator. New individual expression, rough crossover operations in GA and their application examples are described. When a single Flexible Transfer Line manufactures a variety of products, there is a problem to keep a production ratio called production levels. The developed production simulator to have a wide solution search space can solve the problem by a constant individual input. The relation between a production simulator and GA system is that GA system generates some of the input information to operate a production simulator.

Key Words: One-by-one production, Production simulator, Genetic algorithm, Flexible transfer line, Recurring individual

1. Introduction

Many Flexible Transfer Lines (FTL) are working in the world [1]. The single FTL manufactures many kinds of parts. In the FTL, it is important to decide which parts are input into the bay of the production line in order to fit the timing of users' needs and to increase productivity. In order to do this, it is necessary to decide which parts should be input into the bay of FTL. This paper describes the off-line production simulator to decide the sequence.

We have developed the off-line production simulator to do this. This simulator includes GA system and a discrete production simulator. The characteristic of the decision is to keep the parts ratio to be input into the bay. In order to do this, we adopts the new individual expression called a recurring individual.

There is a problem that it takes much time to carry out a single discrete production simulation. As an individual corresponds to the parts sequence, it takes many production simulation times corresponding to the numbers of a population. This is because we do not have a big population size. In order not to lose the individuals' variety, we adopt the method of a constant individual input. We also adopt the rough crossover to reduce the simulation time. We apply the methods to keep individuals' variety for the developed off-line production simulator and ascertained the usefulness or differences of these methods.

2. Production Simulator and GA

GA based production simulator [2] is, first, to generate a population that have n pieces of individuals corresponding to randomly selected genes. The gene corresponds to the parts input sequence. Second is to carry out the production simulations with the input information corresponding to each individual and calculate each individual's fitness by using the simulation results. The third is to carry out the GA operations such as crossover and mutation. The fourth is to evaluate the fitness corresponding to the product efficiency. If the finish condition is not satisfied, the simulator repeats the production simulation by using the individuals in the new generation. If satisfied, the simulation is finished.

In this way, the cycle between the production simulations and the GA operations are carried out and the final result can be acquired.

3. Recurring Individual

The individuals correspond to the parts sequence information of a production simulator. For example, in a case where the variety of parts are A, B, and C, the sequence A,B,C,A,B,C,... or A,A,C,B,A,C,B,... corresponds to the individual. In order to give the equality to a GA problem, it is important that the parts percentage assigned as the gene expressing input sequence is not much different from the goal production ratio. The conventional method to generate individuals does not consider the ratio of each gene in an individual [3][4].

In order to keep the parts input ratio constant, this

paper proposes the new individual expression called a recurring individual. The new individual is expressed as a ring-type individual. The conventional one is like a string [5][6]. The ring-type individual is considered as the first cycle of parts input and is assigned as a tentative locus number "1" and the tentative locus number "δ" which is the last result by continuing to add 1 to the right direction of the ring. The tentative locus number 1 is regarded as the tentative locus number "δ+ 1" corresponding to the start of the second cycle. In the same manner, by recurring the above operation, a single individual is expressed. This individual is called recurring individual. The data described in each locus of the recurring individual corresponds to the part that the operators input in the bay of FTL. In this way, the parts sequence to be input into the bay is expressed with the recurring individual expression as shown in Figure 1.

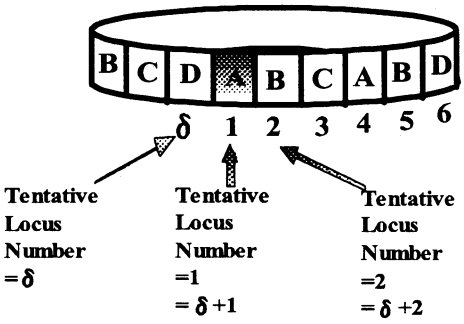


Fig. 1 Recurring individual

4. Crossover to Keep Ratio

The recurring individual has the following characteristic and the above problem can be solved if the characteristic is applied to a crossover. There is the characteristic that gene containing ratio in one cycle of the individual length δ keeps constant whichever locus in one cycle is selected as a tentative locus number 1. By using this characteristic of a recurring individual, the crossover operation algorithm not to change a containing ratio is realized as below.

[Crossover Algorithm to Keep Containing Ratio]

STEP1: Based on fitness, randomly select a pair of recurring individuals as crossover chromosomes and choose $S_{STEP1} = 1$ as loop times.

STEP2: Select one individual from among the individuals selected in STEP1. Call the left end locus of the selected individual Left end number (C_{left}). Randomly select the integer number from 1 to ($\delta - C_{left} + 1$). Call the Lth locus in the selected individual on your right end number (C_{right}). The genes from C_{left} to C_{right} corresponds to the left part of a chromosome to be moved because of a crossover and is called a moving chromosome and the L corresponds to a crossover length. Choose $S_{STEP2} = 1$ as loop times.

STEP3: Find a quantity for each variety of genes located in locuses from C_{left} to (L-1)th. Regards the found quantity for v varieties as α(v).

STEP4: Carry out STEP4-1 ~ STEP4-5 to the individual that is the other individual which is not selected in STEP2 from among a pair of individuals.

STEP4-1: Regard the individual as a tentative locus number 1 and express it as C'_{left}. That is , $C'_{left} = 1$. Also, express a tentative C_{right} as C'_{right} and calculate the value as the following.

$$C'_{right} = C'_{left} + (L-1) \quad \ldots\ldots(1)$$

STEP4-2: Substitute 1 for primitive search time t.

STEP4-3: Find each quantity for v varieties of genes located in locuses from C'_{left} to C'_{right} and regard the quantities asβ(v).

STEP4-4: Carry out the following rule.

[*if*] $\alpha(v) = \beta(v)$ for all of v genes.

[*then*] Regard the values of C'_{left} and C'_{right} as a pair of list and make list = (C'_{left} , C'_{right}). Add the list as one element of crossover candidate lists and continue to STEP5.

[*else*] Renew C'_{left} and C'_{right} by adding 1 to each value of C'_{left} and C'_{right}.

STEP4-5: Add 1 to t and judge the following rule.

[*if*] $t = \delta$

[*then*] Continue to STEP5.

[*else*] Return to STEP4-3.

STEP5: Carry out the following rules on the crossover candidate lists.

[*if*] The crossover candidate lists corresponds to null and the loop times is $S_{STEP1} = S_1$ (optional integer).

[*then*] Repeat from STEP1.

[*if*] The crossover candidate lists correspond to null and loop times is $S_{STEP2} = S_2$ (optional integer).

[*then*] Randomly select C_{left} and crossover length L again and repeat from STEP3. Renew loop times S_{STEP1} by adding 1.

[*if*] The crossover candidate lists correspond to null.

[*then*] Randomly select crossover length L again and repeat from STEP3. Renew loop times S_{STEP2} by adding 1.

[*if*] The crossover candidate lists do not correspond to null.

[*then*] Continue to STEP6.

STEP7: Generate new two individuals by exchanging a moving chromosome and the genes between C'_{left} and C'_{right} selected in STEP6 and finish this algorithm. □

In this way, the crossover operations with recurring individuals can select the starting point of an individual cycle δ times. As a result, it is possible to have a wide search in a candidate set of locus sequences whose gene containing ratios are same.

5. Rough Crossover

If the crossover algorithm mentioned above is used, each time for individual choosing loop, moving chromosome choosing loop and same ratio searching loop becomes tremendous. To solve this problem, the crossover by using similarity (rough crossover) is adopted. The rough crossover strategy is to [1] give a forced crossover even if the containing ratios are not same, [2] give the new individuals by crossover an amendment in order that the two new containing ratios become same.

To carry out the operation of the rough crossover, the following rule is inserted after [then] in *step4-4* of the algorithm described in the previous section ([*] is inserted).

[Rule for rough crossover]
[if] $\alpha(v) \cong \beta(v)$ [then] Generate new two individuals by exchanging a moving chromosome and the genes between C'_{left} and C'_{right} , carry out the algorithm for ratio amendment and finish this algorithm.
Here, $\cong$ means " very close".

The algorithm for ratio amendment is shown below.

[Algorithm for ratio amendment]
A_step1: For each inserted chromosome in the new two individuals, find the part variety "*d*" whose containing ratio number is *g* under the goal ratio and the part variety "*h*" whose containing ratio number is *g* under the goal ratio.
A_step2: For the part variety "*d*", delete *g* from the containing number for *d* . For the part variety "*h*", add *g* to the containing number for *h*. □

For example, the individual A of Fig. 2 has the ratio as A:B:C:D=1:1:0:2. the individual B has A:B:C:D=1:1:1:1. Though the ratios between the individual A and the individual B are not the same, the rough crossover is carried out. The result shows Fig. 3. The new individuals are A' and B'. If the goal ratio is A:B:C:D=4:2:1:3, the ratio of A' is A:B:C:D=4:2:2:2. The ratio of A' is changed to the goal one by changing C to D.

In this way, the rough crossover makes the loops' number reduced.

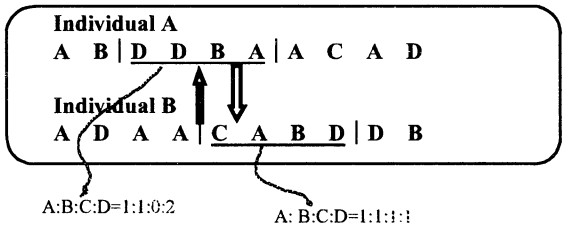


Fig. 2 Rough crossover individuals

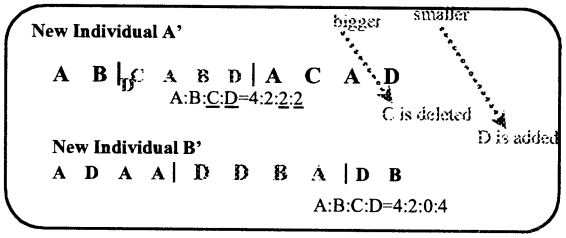


Fig. 3 New individuals

6. Constant Individual Input

There is still a problem that it takes much time to carry out a single discrete production simulation corresponding to one week or one month FTL working in a real world. As an individual corresponds to the parts sequence, it takes many production simulation times corresponding to the numbers of a population. This is because we do not have a big population size. In order not to lose the individuals' variety, we adopt the expectation strategy. In spite of the strategy, we soon converge at a certain individual. In order to keep the individuals' variety, a constant individual input method is adopted.

The method is to input some new individuals every some generations. First, the 5 individuals that have lower fitness among a generation are deleted. Second, the new individuals that are randomly generated are input into the next generation.

8. Application Examples

The developed algorithm is applied for FTL example. The FTL has 8 stations (st_1, st_2, ... , st_8) assembly line and assembles 10 products. Products are input into the station st_1 one-by-one. Assembling times for each variety of products in each station are not same. The goal cycle time of this line is 12 seconds. The production ratio for 10 products is P_1 : P_2 : ..... : P_{10} = 9 : 6 : 7 : 6 : 8 : 7 : 3 : 2 : 4 : 1 . FTL has glitches and they randomly happen every 120 seconds. Their fix time is constant, 20 seconds.

The conditions of GA system are the following; population size = 20, mutation probability = 1%, elite individual = 2, integer constant (a) = 1, individual length (δ) = 53, optional constant (ε) = 865, optional integers S_1 = 5 and S_2 = 5. As the finishing condition of STEP8 in Section3 algorithm, the time when generations = 60 is adopted.

Several simulations by changing a series of a random number were carried out. Table 1 shows the simulation results for individual choosing loop times, moving chromosome choosing loop times and same ratio searching loop times by using the rough crossover. As a comparison, The figures in the

Table 1 Simulation results

Assembly time	Individual choosing loop times	Moving chromosome choosing loop times	Same ratio searching loop times
Time-1	**32,327** (339,874)	**16,492,165** (76,490,301)	**41,651,320** (133,894,654)
Time-2	**29,334** (304,021)	**15,062,155** (70,179,169)	**38,238,654** (122,921,184)
Time-3	**32,185** (367,163)	**16,229,060** (83,103,652)	**40,410,655** (142,157,541)
Time-4	**30,133** (269,443)	**15,320,149** (68,908,023)	**38,735,537** (121,023,554)

parentheses of Fig. 4 show the simulation results by not using the rough crossover. In case of the individual choosing loop times (Time-1), 339,874 of the result by not using the rough crossover is reduced to 32,327. As for the moving chromosome choosing loop times, 76,490,301 is reduced to 16,492,165 and as for the same ratio searching loop times, 133,894,654 is reduced to 41,651,320. In this way, it is ascertained that the rough crossover can make the simulation time much reduced.

Fig. 4 also shows the fitness curves of the simulations results. The bold dotted curve [A] shows the maximum fitness without diversity idea. Around the generation 38, the curve is saturated. Checking up the individuals among a population, it is found that each sequence of the chromosome is very close, the diversity disappears and the simulations go straight to the one local solution. This kind of results is shown in other examples of a series of a random number. The resulted production ratio was $P_1 : P_2 : \ldots\ldots : P_{10} =$ 8.505 : 5.902 : 6.854 : 5.902 : 7.829 : 6.854 : 2.927 : 1.951 : 3.902 : 1 which is very similar to a goal production ratio.The new individual input is carried out every 5 generation.

The bold solid curves [B] show the maximum fitness with diversity idea or to input new individuals. The thin solid curve [B'] shows the average fitness curve. It can be seen that the average fitness curve sometimes drops because of inputting 5 new individuals every 5 generation. Because of it, the solution space is always kept wide and the best fitness curve (bold solid curve) saturated at a higher position.

7. Conclusions

This paper describes the development of an off-line production simulator connected GA system in order to realize one-by-one production for a variety of products. The production simulator includes recurring individuals, their crossover operations and mutation operations not to change the results of a production ratio for each variety of products.

The developed production simulator was applied for a FTL model that assembles a variety of products. As a result, acquired production ratio was very similar to a goal production ratio. The developed production simulator can be used in starting a production plan of FTL that manufactures a variety of products.

REFERENCES

[1] Monden, Y., "Toyota Production System", Institute of Industrial Engineers, Atlanta, GA, 1983.

[2] Yamamoto, H., Simulator of Flexible Transfer Line including Genetic Algorithm and its Applications to Buffer Capacity Decision, Proc. of the 3rd IFAC Workshop on Intelligent Manufacturing Systems, pp127-132, 1995.

[3] Holland, Adaption in Natural and Artificial Systems, Univ. Michigan Press, 1975.

[4] Syswerda, Uniform Crossover in Genetic Algorithm, Proc. Of ICGA89, 1989.

[5] Goldberg, Genetic Algorithm in Search, Optimization and Machine Learning, Addison-Wesley, 1989.

[6] Goldberg, Alles, Loci, and Travelling Salesman Problem, Proc. of ICGA-89, 1989.

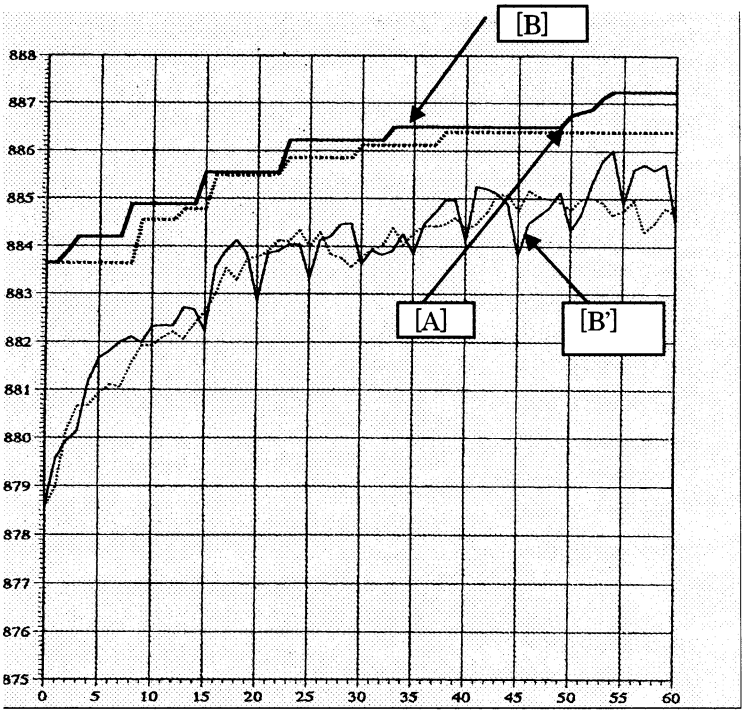


Fig. 4 Fitness curves

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Object Tracking by Using Additive Competitive Learning and Shape Information

Park, Yong-Hun
School of Electrical and Electronic Engineering
Chung-Ang University
221 Heukseok-Dong, Dongjak-Gu, Seoul, Korea
Tel. +82-02-816-8234, Fax. +82-02-816-1856
E-Mail: jdpriest@sirius.cie.cau.ac.kr

Kang, Hoon
School of Electrical and Electronic Engineering
Chung-Ang University
221 Heukseok-Dong, Dongjak-Gu, Seoul, Korea
Tel. +82-02-820-5320, Fax. +82-02-816-1856
E-Mail: hkang@cau.ac.kr

Abstract

In biological vision system, color is a major information as well as edge of an object. In this paper, we propose Additive Competitive Learning model to perform color space classification efficiently, and verify its performance by experiments. Moreover, we show that object tracking in a complex image can be performed easily by system using both color and shape information for real-time pattern recognition.

Keywords: Competitive Learning, Color Space Classification, Object Tracking, Traffic Sign Tracking

1 Introduction

In recent vision system, edge information uses are more than color information uses of object. These methods are showing good performance at simple image, but have difficulties on recognizing objects in moving or complex images like human vision system. There are many researches about human-like vision system model because of this functional limitation. These models are stereo vision system [1], active vision system [2], shape recognition system [3], etc.

In these vision system, image segmentation is essential process as basic pre-processing. The methods of image segmentation could be categorized into methods for dividing an image space [4] and those for clustering a feature space derived from an image [5].

The former methods could accomplish simply and efficiently at one dimensional space classification, but could not determine proper cluster boundary at upwards of two dimensional space because shape of clusters is rectangular like shown in Figure 1 (a). But the latter methods are suitable at upwards of two dimensional space because it can coordinate shape of clusters like shown in Figure 1 (b) [6].

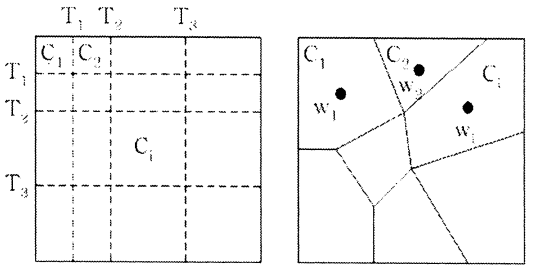


Figure 1. Two dimensaional space classification
(a) Thresholding (b) Competitive learning

And, the methods of image segmentation could be categorized into point, line, shape, boundary and color detection by its purpose. In those, color is the most particular information of all informations, so performance of vision system is greatly influenced by performance of color detection.

In this paper, we propose Additive Competitive Learning model to perform color space classification efficiently, and verify its performance by experiments. Moreover, we show that object tracking in a complex image can be performed easily by system using both color and shape information for real-time pattern recognition. Figure 2 is the system structure using this paper.

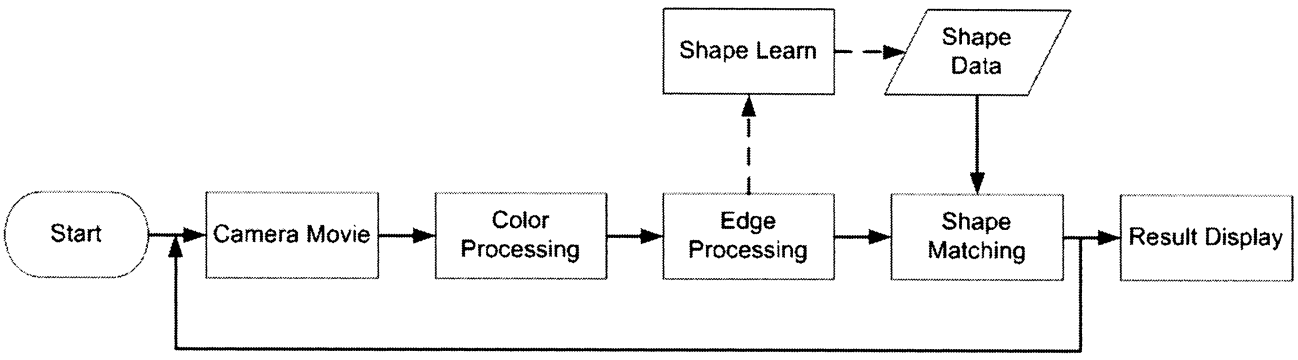


Figure 2. Object Tracking System Structure Using in This Paper

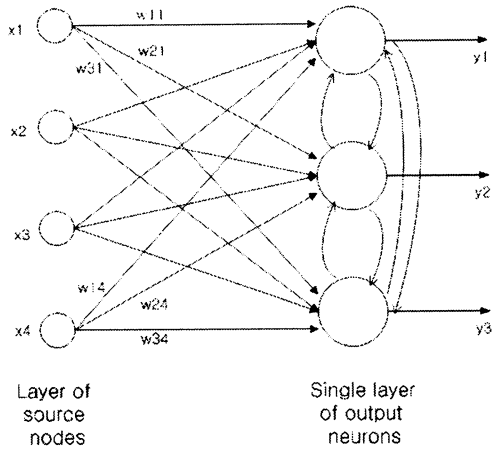


Figure 3. Structure of Competitive Neural Network

The next section proposes Additive Competitive Learning model improved relatively more than existing competitive learning model and explain color space classification method by using proposed model. The third section explains shape mathing method. The forth section shows a experimental result, and the last section concludes this paper.

2 Color Space Classification by Using Additive Competitive Learning

2.1 Competitive Learning

Competitive neural network structure is shown in Figure 3. In this competitive network, only one output neuron is active through compete each others.

This competitive learning algorithm is progressed as follows. First, compute net value of each output neuron, and then decide output neuron as winner neuron with highest net value. Second, set 1 at real output of this winner neuron, and set 0 at real output of the others. Let N, x_{ji}, net_j and w_{ji} be number of input nodes, input value into ith input node, net value of jth output node in M output nods and connection weight between ith input node and jth output node. Equation (1) and (2) is net value(net_j) and output(y_j) of jth output neuron by competition.

$$net_i = \sum_{i=1}^{N} w_{ji} x_i \tag{1}$$

$$y_i = \begin{cases} 1 & \text{if } net_j > net_k \quad \text{for all } j, j \neq k \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Winner neuron is only one that permitted learning about input vector $\mathbf{x}(=\{x_1, x_2, x..., x_N\})$. Equation (3) is weight update formula for winner neuron.

$$w_{ji} = \begin{cases} w_{ji} + \eta(x_i - w_{ji}) & \text{if neuron j wins} \\ w_{ji} & \text{if neuron j loses} \end{cases} \tag{3}$$

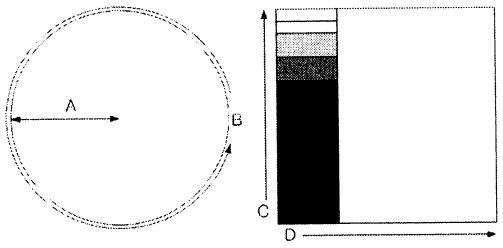


Figure 4. HIS Color Model

A. Saturation B. Hue C. Intensity D. All Hues

η is the learning coefficient ($0 < \eta << 1$) in these equation. In general, there are many uses of learning coefficient from 0.01 to 0.3.

After learn from all input vectors, weight vectors become a criterion for divide each cluster as place at each center of clusters.

2.2 Color Space Selection

It is very important that select color model in vision system. There are many uses of RGB, CMY, CMYK, HSI color models.

In general, images are represented by RGB color model in computer. This model is easy to realize to hardware and represent colors. Moreover, there are many uses because of easiness of concept. But, there is a weak point that difficult to represent the tone of color. Moreover, there are some changes of color according to variation of lightning and it is difficult to split by hue because hue is dispersed in RGB color space. On the other hands, it is possible that free of shadow and lightning by using HSI color model. And it is easy to split color space than other color models because HSI color space is distributed by hue.

Each elements of HSI color model have a meaning in Figure 4. Hue is distribution of a color tone like yellow and red, saturation is depth of hue and Intensity is degree of lightning.

Most image input devices adopt RGB color model ordinary, so processing that converse RGB color model to HSI color model is requested to use HSI color model. It is next equation (4), (5), (6) that convert from RGB to HSI color model.

$$H = \begin{cases} \theta & \text{if } B \leq G \\ 360 - \theta & \text{if } B > G \end{cases} \tag{4}$$

with

$$\theta = \cos^{-1}\left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{1/2}} \right\}$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R,G,B)] \tag{5}$$

$$I = \frac{1}{3}(R+G+B) \tag{6}$$

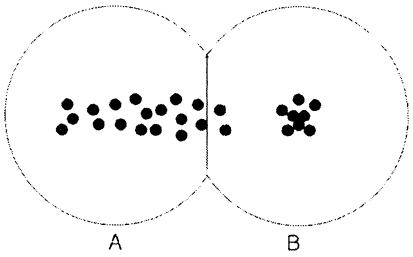


Figure 5. Classification of Uncertain Area

2.3 Color Space Classification by Using Additive Competitive Learning

When to use the existing competitive learning, it is possible that wrong classification is performed near boundary of each cluster. In Figure 5, there occurs a case that the area of cluster A is classified to cluster B. Because vectors have an uncertainty in space, these cases occur.

In this paper, we propose Additive Competitive Learning that increase numbers of cluster for classify uncertain area efficiently. After deferring classification of uncertain area by limit size of clusters, create a new cluster at the uncertain area. And then, the new cluster is combined to a closer one between cluster A and B. With this method, it is possible that wrong space classification can be avoided, and complex space classification is possible.

Next algorithm is the color space classification method proposed in this paper.

※ Color Space Classification Algorithm

1. Let N and M be the number of input nods and the number of output nods respectively. Set initial weight vectors arbitrary and cluster radius r for limit cluster area.
2. Decide the winner neuron by competition.
   2-1. If spatial distance between winner weight vector and input vector is more far than radius r:
   Create a new weight vector (w') at the same location with the input vector (x_{new}). Radius r' of new cluster that creates by new weight vector is determined by next equation.

$$w' = x_{new}$$

$$r' = (\| w_m - w' \| - r_m + \| w_n - w' \| - r_n)/2$$

   After learning is completed, this cluster is combined to cluster of winner weight. Go to step 2.
   2-2. If spatial distance between winner weight vector and input vector is more near than radius r or same with r:
   After weight update, go to step 2.
3. After repeat K times, Stop learning.

If it is same the size of cluster by new weight vector as existing cluster, there is a possibility that violate other cluster area. So, it must have a limitation in new cluster size. In Figure 6, since radius of new cluster must farther $\| w_m - w' \| - r_m$ and closer than $\| w_n - w' \| - r_n$, we decide the radius r' as the middle value between them.

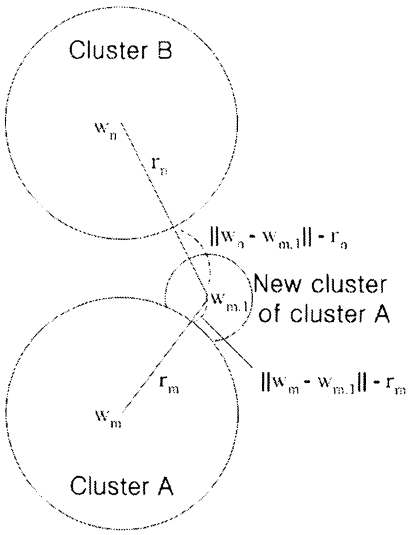


Figure 6. Creating New Cluster of Proposed Algorithm

3 Shape Matching for Object Tracking

We used simple shape matching method for tracking arbitrary object. After memorize 100x100 pixel size picture priviously, measure euclidean distance between memorized picture and target picture as shown in figure 7. If this euclidean distance is lower than specific size, we decide that there exists the target object.

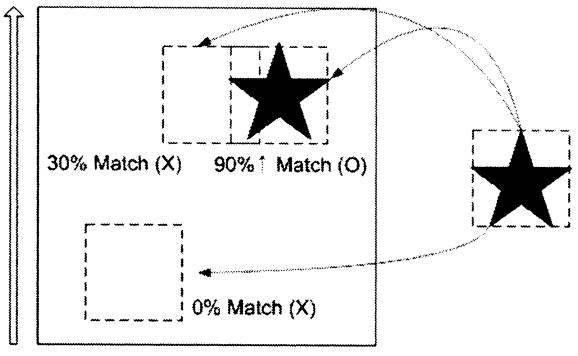


Figure 7. Shape Matching for Object Tracking

4 Experimental result

For experiments, we used moving pictures those were 320x240 resolution and 24fps(frame per second) from CCD camera for PC.

We performed experiments that tracking a traffic sign pattern and star shape object by using Additive Competitive Learning and simple shape matching algorithm. Figure 8, 9 shows the experimental results. Because color space classification is performed properly, we could track objects by using very simple algorithm described in section 3. And we performed a oudoor experiment with this system. This experimental result is shown in figure 10.

All experiments were available by real-time.

5 Conclussion

In this paper, we proposed the color space classification method by using Additive Competitive Learning to realize vision system like human. By using proposed Additive Competitive Learning, it is possible that wrong space classification can be avoided. And, it could create complex shape clusters. Moreover, through experiments, pattern recognition problem can be simplified by reason that efficient classification of color space. Therefore we may realize complex and real-time vision system.

Acknowledgement

This work was supported by grant no. M10107020001-02B2202-00110 from the Ministry of Science & Technology (MOST) of Korea.

References

[1] Gavrila, D. M. & Franke, U. & Wohler, C. & Gorzig, S. (2001), Real Time Vision for Intelligent Vehicles, IEEE Instrumentation & Measurement Magazine, Vol. 4, No. 2, pp.22-27

[2] Miura, J. & Kanda. T. & Shirai, Y. (2000), An Active Vision System for Real-Time Traffic Sign Recognition, IEEE Intelligent Transportation Systems Proceedings, pp.52-57

[3] Androutsos, D. & Trahanias, P.E, & Venetsanopoulos, A.N. (1997), Application of Active Contours for Photochromic Tracer Flow Extraction, IEEE Transactions on Medical Imaging, Vol. 16 No. 3, pp.284-293

[4] Baraldi, A. & Parmiggiani, F. (1996), Single Linkage Region Growing Algorithms Based on The Vector Degree of Match, IEEE Transactions on Geoscience and Remote Sensing, Vol. 34 No. 1, pp.137-148

[5] Xiao-Ping Zhang & Desai, M.D. (2001), Segmentation of Bright Targets Using Wavelets And Adaptive Thresholding, IEEE Transactions on Image Processing, Vol. 10 No. 7, pp.1020-1030

[6] Toshio Uchiyama & Michael A. Arbib (1994), Color Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 12, pp.1197-1206

Figure 8. Tracking of Traffic Sign Pattern

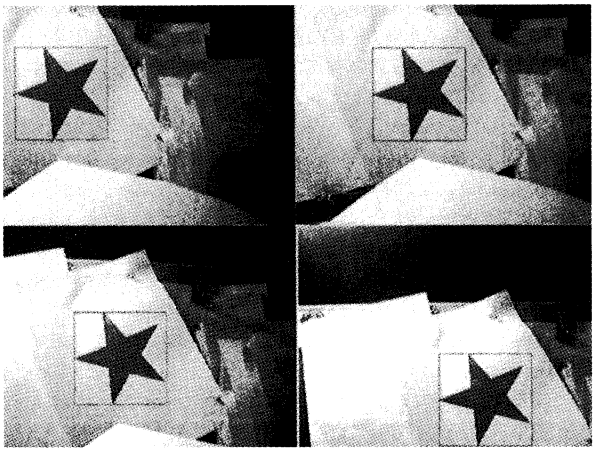


Figure 9. Tracking of Star Pattern

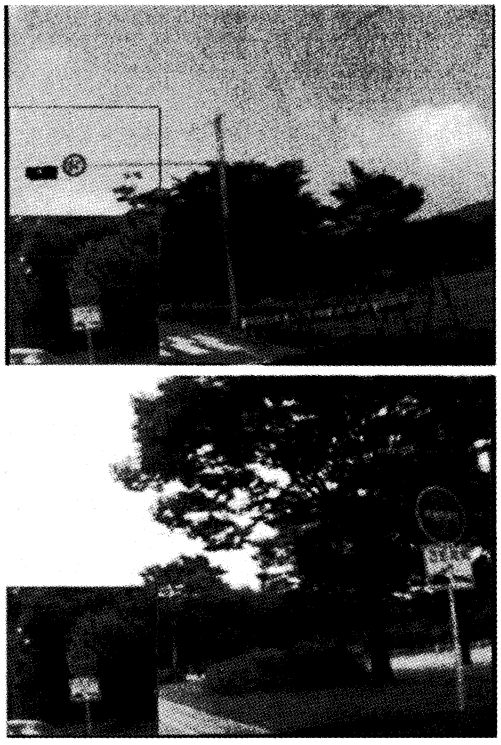


Figure 10. Tracking of Outdoor Trafic Sign

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

The Analysis for the Path of an Object in Earth Orbit with Genetic Algorithms

Ryuji Goto
Graduate School of IT Professional Course,
Hosei University:1914-7, Kosuzume-cho,Totsuka-ku,Yokohama-shi 244-0004, Japan
E-mail: rgoto@paw.hi-ho.ne.jp

Yuji Sato
Faculty of Computer and Information Science,
Hosei University:3-7-2, Kajino-cho, Koganei-shi
Tokyo 184-8584, Japan
E-mail: yuji@k.hosei.ac.jp

Abstract

Issues such as multiobjective optimization, time-series prediction, the analysis of noisy observation data, and the solution of implicit functions are all crucial in the consideration of real world problems, and research into the applicability of evolutionary computer techniques to these problems has already begun [1–9]. However, there are only a few examples of studies where evolutionary computer techniques have been applied to problems that involve all of these issues at the same time. One such example is a previous study in which we reported on the effectiveness of genetic algorithms as a tool for tracking objects as they move towards a target while making evasive maneuvers in order to avoid pursuit or attack.

In this paper, we verify the applicability of genetic algorithms to the more complex problem of analyzing the paths of objects in the earth orbits. **Key words are** Orbit, Tracking , 7-Path-Eelements, Analysis-from-Noisy-Data.

1 Introduction

In previous report, we considered the 2-dimensional movement of an object whose evasive motion was assumed to consist of constant-velocity straight-line, simple sinusoidal and sawtooth motion[10, 11]. In this paper, we aim the analysis for 3-dimensional path followed by moving objects around the earth. The path consist of circles, ellipses, parabolas and hyperbolas depending on their velocity and distance from the earth at the time they are injected into orbit. When the distance and velocity are constant functions, the moving object describes a circle centered on the earth's center of gravity, and if the velocity is made any higher then it performs elliptical motion with a focal point at the earths' center of gravity. Increase the velocity further still and its path becomes a parabola with its focal point at the earth's center of gravity, and if the velocity increases still more then the moving object describes a hyperbola and escapes from the earth's gravitation. In this paper we will concentrate on elliptical motion and circular motion.

When the distance vector and velocity vector of a object in orbit around the earth are known, its orbiting path elements can be calculated according to the laws of Kepler and Newton [12]. However, it is necessary to compensate for factors that disturb Kepler's laws, such as the gravitational forces of other celestial objects, the earth's equatorial bulge, and frictional forces exerted by the upper atmosphere. On the other hand, when the distance vector and velocity vector of a moving object are unknown, such as a newly discovered moving object, it is difficult to mathematically determine all the orbiting path elements of the moving object only from time-series observation data measured by one observation equipment on the earth. This is because the observation data includes various types of errors that are difficult to predict, such as wavelength fluctuations, observation equipment errors and conversion errors, and because the orbit of the moving object is a complex implicit function of seven different orbiting path elements (longitude of ascending node, inclination, argument of perigee, velocity of perigee, semi-major axis, ratio of axis, and true anomaly).

In this paper we aim to verify by way of computer simulation that genetic algorithms can be used to analyze all seven of these orbiting path elements at the same time from noisy time-series observations of an unidentified object's distance, bearing and elevation obtained from terrestrial observation equipment. If it is possible to calculate these orbiting path elements only from time-series data obtained from one terrestrial observation equipment, then it is possible to detect when a third party has intentionally altered the course of an object while in orbit and we can get the simplest orbit tracking system.

2 Tracking Three Dimensional Orbit of an Object in the Space

2.1 Geocentric Coordinate and the Orbiting Path Elements

The relationship between geocentric coordinates and the orbiting path elements of a moving object is illustrated in Figure 1. This shows an example where the moving object describes an elliptical (or circular) orbit, which is determined by its velocity and distance from the earth at the time it is injected into orbit. The orbital plane always includes the earth's center of gravity. A geocentric coordinate system can be established with the X axis pointing towards the vernal equinox point on the equatorial plane, a Y axis at right angles thereto, and a Z axis pointing towards the earth's north pole. Coordinates based on this system are called inertia coordinate (XYZ). We will assume that observation equipment (ob) situated on the earth (E) is used to make periodic observations of the object's distance, bearing and elevation as seen from the observation point. The observed position of the moving object when it is first discovered at time $t0$ is referred to as the $t0$-position shown in Figure 1. The observed position at time tn is called the tn-position.

These positions observed by the observation equipment (ob) can be directly converted into the observed distance ($oDt0$, $oDtn$), observed bearing ($oBt0$, $oBtn$) and observed elevation ($oEt0$, $oEtn$) based on earth

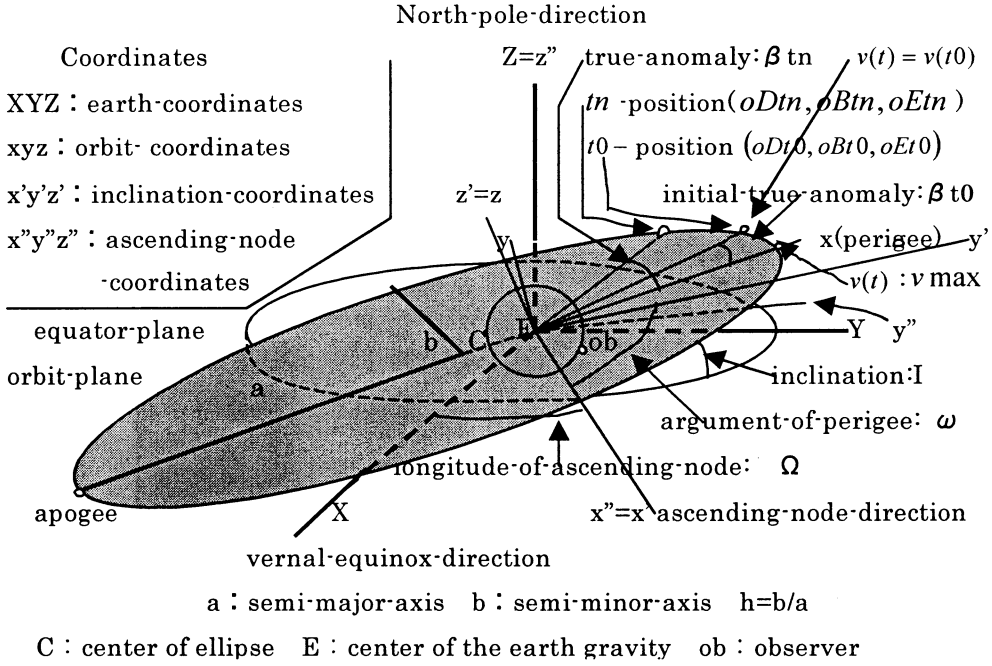


Figure1: The relationship between geocentric coordinates and the orbiting path elements of a moving object. This shows an example where the moving object describes an elliptical orbit.

coordinates (*XYZ*) by correcting for the installed location of the observation equipment, for positional factors relating to the earth's rotation, and for the passage of time [12]. These observed values include noise caused by errors in the observation system due to factors such as wavelength and light beam fluctuations, instrumentation errors and conversion errors. Accordingly, the problem to be solved is the problem of determining the seven orbiting path elements of a moving object based on noisy time-series data on the observed distance, bearing and elevation of the moving object, as shown in Figure. 1. These orbiting path elements are the longitude of ascending node (Ω), inclination (I), argument of perigee (ω), velocity of perigee ($v$max), semi-major axis (a), ratio of axis (h) and initial true anomaly ($\beta t0$).

2.2 Formulation of the Orbital Path Analysis

In the following, we show the relationship between the inferred values of the orbiting path elements— longtude of ascending node (Ω), inclination (I), argument of perigee (ω), velocity of perigee ($vmax$), semi-major axislongtude of ascending node (Ω), inclination (I), argument of perigee (ω), velocity of perigee ($vmax$), semi-major axis (a), ratio of axis (h) and initial true anomaly ($\beta t0$) — and the moving object's distance (Dtn), bearing (Btn) and elevation (Etn) at time tn.

Figure 2 shows the elliptical orbit of an object moving in the yz plane of the coordinate system(xyz) of the moving object's orbit. The origin of this coordinate system is at the earth's center of gravity E. The moving object describes an ellipse with one focal point at the origin — i.e., the earth's center of gravity E.

The ratio of the distance between the ellipse's center C and the earth's center of gravity E relative to the semi-major axis (a) is called the eccentricity ε. This value is defined as $\varepsilon = \sqrt{1-h^2}$. Accordingly, the central coordinates C of the ellipse are $(-a\sqrt{1-h^2}, 0, 0)$.

This elliptical orbit can be expressed in the form of Equation (1).

$$\frac{y^2}{b^2} + \frac{\left(x + a\sqrt{1-h^2}\right)^2}{a^2} = 1 \quad \text{.....(1)}$$

The value of v(t) has graduating velocity, is maximum value (vmax) at perigee and minimum value at apogee. The relation between $v(t), v(t0)$ and $v$max is expressed in Equations of Figure.2. The initial true anomaly at the time of the initial observation t0 is denoted by $\beta t0$, and the true anomaly at time tn is denoted by βtn.

The relation between $\beta t0$ and $\theta t0$ is expressed in Equations of Figure 2.

This relation is the same as between βtn and θtn.

The length arc_{t0-tn} of the arc through which the moving object travels between time $t0$ and time tn is expressed by the elliptical integration formula shown in Equation (2). The value of θtn is determined by evaluating Equation (2).

The coordinates in the ellipse corresponding to θtn are (xtn,ytn, 0).

$$arc_{t0-tn} = \int_{t0}^{tn} v(t)\,dt = \int_{\theta t0}^{\theta tn} a\sqrt{1-\left(1-h^2\right)\cos^2\theta}\,d\theta \quad \text{..(2)}$$

In Figure 2, the angle αtn corresponding to the point P where the circle with radius of a (dotted line) intersects with the line $x = xtn$ is given by Equation (3).

$$\alpha tn = \tan^{-1}\left((\tan\theta tn)/h\right) \quad \text{.....(3)}$$

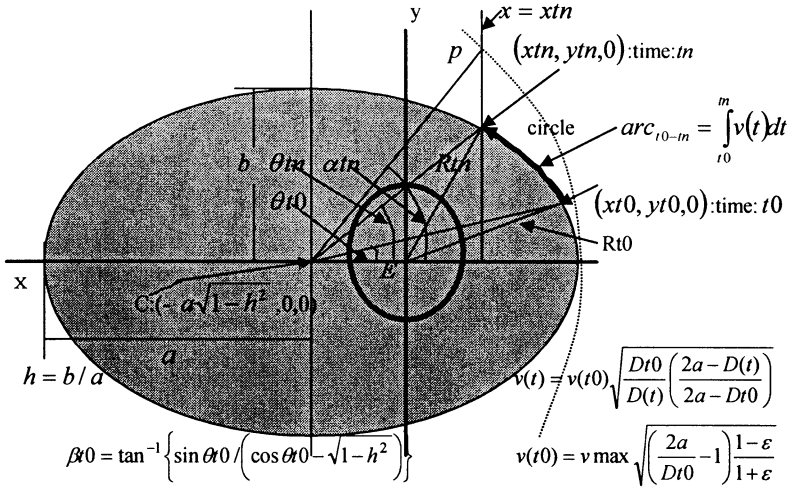


Figure 2: Orbital plane. This shows the elliptical orbit of an object moving in the yz plane

Thus the coordinates (xtn, ytn, ztn) of the moving object at time *tn* are shown in Equations (4), (5) and (6).

$$xtn = -a\sqrt{1-h^2} + a\cos\alpha tn \quad \text{.....(4)}$$

$$ytn = a\,h\sin\alpha tn \quad \text{.....(5)}$$

$$ztn = 0 \quad \text{.....(6)}$$

The *xy* plane of the *xyz* coordinate system is rotated clockwise about the *z* axis through an angle equal to the argument of perigee (ω) to convert it into an inclined coordinate system ($x'y'z'$). This coordinate transformation is expressed by Figure 3 and Equations. (7), (8) and(9).

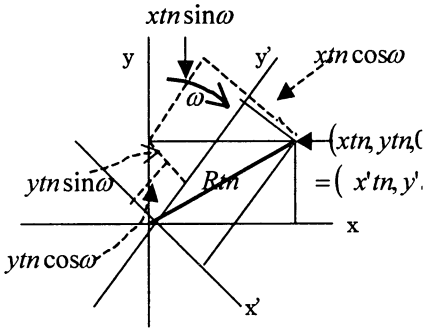


Figure 3: Conversion orbit to inclination

$$x'tn = xtn\cos\omega - ytn\sin\omega \quad \text{.....(7)}$$

$$y'tn = xtn\sin\omega + ytn\cos\omega \quad \text{.....(8)}$$

$$z'tn = ztn = 0 \quad \text{.....(9)}$$

The $y'z'$ plane of this $x'y'z'$ coordinate system is rotated clockwise about the *x* axis through an angle equal to the inclination (*I*) to convert it into ascending node coordinate system $x''y''z''$. This coordinate transformation is expressed by Equatins(10),(11) and (12),is the same idea as ω of figure 3.

$$x''tn = x'tn \quad \text{.....(10)}$$

$$y''tn = y'tn\cos I - z'tn\sin I \quad \text{.....(11)}$$

$$z''tn = y'tn\sin I + z'tn\cos I \quad \text{.....(12)}$$

The ascending node coordinates ($x''y''z''$) are rotated in the $x''y''$ plane (the earth's equator plane) about the z'' axis through an angle equal to the longitude of ascending node (Ω) to convert them into earth coordinates (*XYZ*). This coordinate transformation is expressed by Equations (13), (14) and (15), is the same idea as ω of figure 3.

$$Xtn = x''tn\cos\Omega - y''tn\sin\Omega \quad \text{.....(13)}$$

$$Ytn = y''tn\cos\Omega + x''tn\sin\Omega \quad \text{.....(14)}$$

$$Ztn = z''tn \quad \text{.....(15)}$$

The distance (*Dtn*), bearing (*Btn*) and elevation (*Etn*) of the moving object in earth coordinates (*XYZ*) can thus be expressed as shown in Equations (16), (17) and (18).

$$Dtn = \sqrt{Xtn^2 + Ytn^2 + Ztn^2} \quad \text{.....(16)}$$

$$Btn = \tan^{-1}(Ytn/Xtn) \quad \text{.....(17)}$$

$$Etn = \tan^{-1}\left(Ztn/\sqrt{Xtn^2 + Ytn^2}\right) \quad \text{.....(18)}$$

Accordingly, the problem addressed in this paper — i.e., that of analyzing the three-dimensional orbit of a moving object — can be formulated as an inverse problem involving complex implicit functions where it is necessary to find the seven orbiting path elements of a moving object by working backwards from noisy time-series observations of its distance, bearing and elevation obtained from terrestrial observation equipment.

3 Method to Apply Genetic Algorithms

3.1 The Orbiting Path Elements Determined by Genetic Algorithms

The seven orbiting path elements — longitude of ascending node(Ω), inclination(I), argument of perigee (ω), velocity of perigee ($v max$), semi-major axis (a), ratio of axis (h) and initial true anomaly ($\beta t0$) — constitute a complex implicit function, so we will try to use genetic algorithms to determine their values. In exact genetic analysis of this simulation, the velocity of initial $v(t0)$ is analyzed instead of v max for simplification. The v max is determined from $v(t0)$ analytically through relational Equations in Figure 2.

3.2 Chromosome Coding Method

We defined chromosomes respectively corresponding to each of these seven orbiting path elements, and we expressed a single individual as a set of these chromosomes. Each chromosome consist of different length bits according to range of its value and necessary resolution. Semi-major-axis(a) is 21 bits, velocity of initial($v(t0)$) is 17 bits, longitude of ascending node(Ω),inclination(I), true argument of perigee (ω) and initial perigee angle($\theta t0$) are 28 bits, ratio of axis(h) is 23 bits. LSB (least significant bit) is corresponding to the necessary resolution. The minimum units of these orbiting path elements were set as follows: semi-major axis: 0.9192m, velocity of initial:0.0763m/s, $\Omega, I, \omega, \theta t0$: 3.3528×10^{-7} deg, ratio of axis: 1.192×10^{-7}.

The physical range over which each chromosome was expressed was set considering the range of possible orbits of the moving object and the observation precision assuming laser observations (i.e., a distance error of a few tens of m, and an angular error of a few μradians).

In exact physical calculation of this simulation, semi-major-axis(a) is biased by constant value 6400,000m that is corresponding to the radius of the earth.

3.3 Fitness Function

We determine the distance error between the observed position data of the moving object and the position data inferred by GA, and we defined the fitness of an individual based on the reciprocal of the square root of the sum of squares of these errors. The fitness function is shown in Equation (19).

$$f = k(n+1)/\{\sum_{i=0}^{i=n}\sqrt{Xdf(ti)^2 + Ydf(ti)^2 + Zdf(ti)^2}\ \} \cdots\cdots(19)$$

Where

$Xdf(ti)=oXti-esXti, Ydf(ti)=oYti-esYti, Zdf(ti)=oZti-esZti$

Here, *oXti*, *oYti* and *oZti* are the *X*, *Y* and *Z* axis components of the moving object position obtained from the observed data at time *ti*, *esXti*, *esYti* and *esZti* are the *X*, *Y* and *Z* axis components of the moving object position estimated by GA at time *ti*, $n+1$ is the number of observations, and k is a suitable constant. For example, when $k=1$, it is set so that $f=1.0$ when the average distance difference is 1 km.

3.4 Genetic Operation

As the method for selecting the group of individuals carried forward to the next generation from the current generation, we retained a fixed proportion of the fittest individuals, and the number of discarded individuals is supplemented by roulette selection to preserve the total number of individuals. Specifically, if M is the number of discarded individuals, then $M/2$ pairs of individuals are chosen by roulette selection from the N individuals of the current generation. The $M/2$ pairs of individuals are then subjected to single-point crossovers and spontaneous mutations to produce M individuals to make up the deficit.

These crossovers and spontaneous mutations are performed independently on all of the chromosomes for the true argument of apogee, velocity of initial, semi-major axis, ratio of axis and initial true anomaly which constitute a single individual. In preliminary trials we found that the solution starts to converge after the 15th generation, so for subsequent generations we restricted the crossover positions so that — for P individuals out of the above mentioned M individuals — a number of chromosome units constituting the individuals were exchanged instead of performing ordinary single-point crossovers. We did this to avoid destroying orbiting path elements that had already approached convergence by applying normal single-point crossovers uniformly to all the chromosomes. Also, our preliminary trials showed that there was a certain degree of interdependence between different chromosomes of an individual corresponding to orbiting path elements (such as $\beta t0$ and ω or a and h), so once an individual has approached convergence it is better to perform single-point crossovers involving the exchange of several chromosomes as a set.

4 Evaluation Tests

4.1 Evaluation Method

We made a software system for the evaluation of this research. This system consists of three modules of software, Observed data Generator, Estimated value Generator and Estimated value Evaluator. They do cooperative works for evaluation.

Observed data Generator

Based on the orbiting path elements of a moving object provided by the operator, the observed data generator calculates the theoretical values of the object's distance (*Dti*), bearing (*Bti*) and elevation (*Eti*) at time *ti* for 2-second intervals from the time $t0$ at which the observation starts. It then adds normal random number errors εdti, εbti and εeti to the calculated theoretical values to simulate the errors produced by an observation system, such as wavelength and light beam fluctuations, instrumentation errors, and conversion errors, to produce the observed distance (*oDti*), observed bearing (*oBti*) and observed elevation (*oEti*). As the observation time ti is increased, the observed data generated for experimental use are stored along with the observation time ti in a database for observed data. This observation data gathering cycle is continued until simulation ends.

Estimate value Generator

The Estimated value Generator generates the estimated distance (*esDti*), estimated bearing (*esBti*) and estimated elevation (*esEti*), and uses genetic algorithms to update the estimated values of the orbiting path elements.

First, *esDti*, *esBti* and *esEti* ($t0 \le ti \le tn$) are calculated from Equations (16), (17) and (18) based on the values of the orbiting path elements corresponding to the constituent chromosomes of the individuals in the genetic operations. The initial values of the orbiting path elements—i.e., the initial values of the chromosomes—are randomly set to values in the defined ranges by using uniform random numbers. The values of *esDti*, *esBti* and *esEti* ($t0 \le ti \le tn$) are calculated by genetic operations at each generation, and these values are sent to the Estimated value Evaluator. Next, we will describe how the genetic algorithm is used to update the estimated values of the orbiting path elements. In the Estimated

value Generator, the values of *esDti,esBti* and *esEti* ($t0 \leq ti \leq tn$) corresponding to each individual are sent to the Estimated value Evaluator, and the corresponding fitness values are received from the Estimated value Evaluator.

Next, based on the received fitness values, the method described in section 3.4 is used to select the fittest individuals and perform crossovers and spontaneous mutations, thereby updating the generation, i.e., updating the estimated values of the orbiting path elements.

Here, the updating of the estimated values of the orbiting path elements using a genetic algorithm is started at the point when the 201st set of observed data is stored in the database for observed data (i.e., 400 seconds after the observations are started). In our simulation, the parameters of the genetic operations were set as follows: total number of individuals:1800, number of elite individuals:60, number of individuals with limited crossover positions:300 (only from 15th generation onwards), crossover ratio:0.8, spontaneous mutation rate: 0.001 for the first generations (in order to broaden the search space), and 0.0005 from the 15th generation onwards (when the solution starts to converge; based on previous experiments).

Estimated value Evaluator

For every observation time tn–$t0$, the Estimated value Evaluator calculates the fitness of each individual according to Equation (19) based on the values of the observed distance (*oDti*), observed bearing (*oBti*) and observed elevation (*oEti*) input from the database for observed data, and the estimated distance (*esDti*), estimated bearing (*esBti*) and estimated elevation (*esEti*) input from the Estimated value Generator. These calculated fitness values are sent to the Estimated value Generator, where they are used for genetic manipulation.

The above processes of generating observed values, generating estimated values and performing evaluation are repeated until a stopping criterion is met. The stopping criterion was taken to be the fulfillment of either of two conditions: that an individual appears whose fitness exceeds a preset standard fitness, or that the number of generations of genetic manipulation becomes greater than 50.

4.2 Exmental Results

Figure4(a) shows how the maximum and average values of the fitness of individuals vary with the number of generations in cases where the movement of the moving object is assumed to be elliptical, and Figure4(b) shows the corresponding relationships in cases where the moving object has a circular orbit. To obtain a maximum fitness value of 1.0, the fitness *fg* on the vertical axis in these figures is the normalized value obtained from the relationship $fg = 1 - 1/f$, where f is the fitness defined in Equation (19). When observations are made at 2-second intervals with observation equipment having a distance observation precision of about 20 m and an angle observation precision of about 3 μradians, the orbiting path element analysis errors for an elliptical orbit were as follows: longitude of ascending node (Ω) and inclination (I): 0.003°, argument of perigee (ω) and initial true anomaly ($\beta t0$): 0.3°, velocity of perigee ($vmax$): 0.1%, semi-major axis (a): 0.3%, ratio of axis (h): 0.06%.

These values indicate the precision of the results obtained when investigating the values of the orbiting path elements corresponding to the convergence values of the maximum degree of fitness, and are the average values obtained after repeating the test 20 times.

The tracking time up to the 30th generation when the solution converged from the initial observation was about 9 minutes, and the distance traveled by the moving object during this time was about 10% of the ellipse circumference. The analysis error of the orbiting path elements for a circular orbit was the same as an elliptical orbit for the longitude of ascending node (Ω) and inclination (I), while the errors in the argument of perigee (ω) and initial true anomaly ($\beta t0$) were 0.2°, the velocity of perigee($vmax$)was 0.08%, semi-major axis(a) was 0.2%,and the ratio of axis (h) was 0.01%.

It was thus possible to determine the solution with a higher degree of accuracy than for an ellipse.

4.3 Discussion

From the above experimental results, it can be judged that by applying genetic algorithms to the orbit analysis of a moving object whose elliptical orbiting path elements are all unknown, it is possible to simultaneously analyze

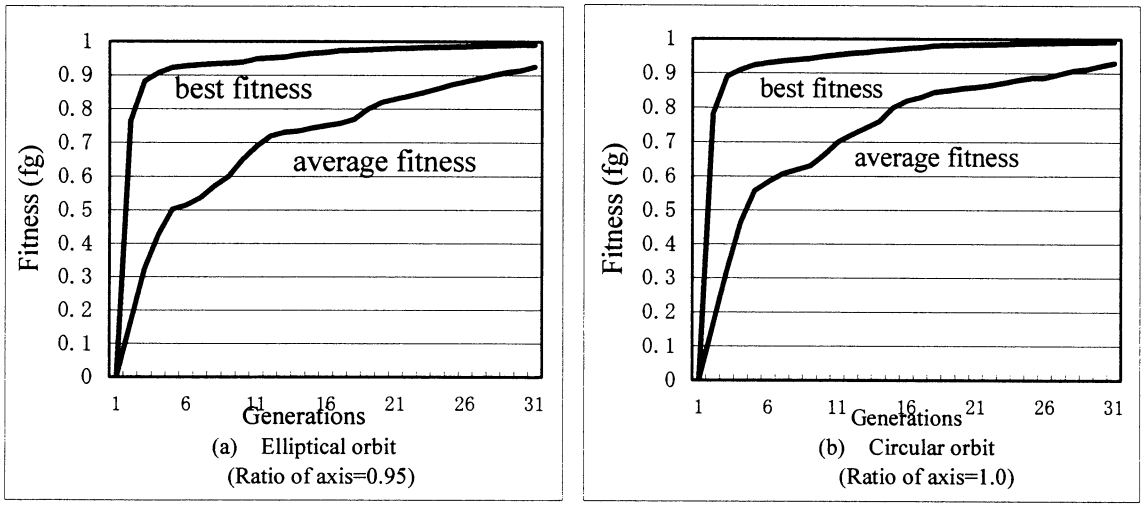


Figure4: This shows how the maximum and average values of the fitness of individuals

all the orbiting path elements only from time-series values of the observed distance, bearing and elevation of the moving object obtained from observation equipment.

Since the analysis results are all based on observation data, the solution consists of values that incorporate the effects of perturbations. This technique thus has the advantage that over short periods there is no need for the perturbation corrections necessary for orbit analysis based on the laws of Kepler and Newton. The average error in the estimated position of a moving object in an elliptical orbit is 1.3 km, and the error in the estimated transit time is about 0.1%. Also, the time taken to converge on a solution after the initial detection is about 9 minutes, and it can thus be judged effective as an orbit estimation system. If it is possible to calculate the orbiting path elements only from time-series data obtained from one terrestrial observation equipment, then it is possible to detect at an early stage when a third party has intentionally adjusted the path of a moving object in mid-orbit. A circle can be regarded as a special case of an ellipse where the analysis result of the ratio of axis (h) is 1.0. It is found that the maximum degree of fitness converges slightly faster for circular orbits than for elliptical orbits. This seems to be because circles have no perigee or apogee, so the value obtained by adding the argument of perigee (ω) and initial true anomaly ($\beta t0$) can be regarded as a single unknown number, so for practical purposes there is one less unknown quantity to be determined. Subjects requiring further study include investigating the effects of observation errors on the solutions, and making a more detailed investigation of factors such as the overall number of individuals, the number of elite individuals, the crossover rate and the spontaneous mutation rate in order to improve the accuracy of the results. The analysis method used in this paper can also be applied to parabolic and hyperbolic orbits as well as elliptical orbits, so it would also be worthwhile investigating the possibility of analyzing parabolic and hyperbolic orbits in practical amounts of time and with practical levels of precision.

5 Conclusion

As a problem including various issues such as multiobjective optimization, time-series prediction, the analysis of noisy observation data, and the solution of implicit functions, we have investigated the applicability of genetic algorithms to the analysis of the three-dimensional and graduating velocity motion of an object in earth orbit. using a computer to simulate the data produced by one terrestrial observation equipment when measuring the position of a newly detected unidentified moving object whose orbiting path elements are all unknown, we obtained noisy time-series information on the object's observed distance, bearing and elevation.

Based on this simulated data, we have shown that genetic algorithms can be used to determine the orbiting path elements of the moving object to a practical degree of precision in a practical amount of time.

In the future ,we aim to demonstrate that genetic algorithms are suitable for solving three-dimensional and graduating velocity analysis problems for objects moving in space by expanding this technique to include the analysis of parabolic and hyperbolic orbits in a practical amount of time and with a practical level of precision.

Also, we aim to demonstrate that genetic algorithms are suitable for the more wide and complex analysis of the facts in the noisy data. It must lead to the analysis of the artificial life etc.

References

[1] Oakley, H.: Two Scientific Applications of Genetic Programming: Stack Filters and Non-Linear Equation Fitting to Chaotic Data. in K.E. Kinnear, Jr (ed.), Advances in Genetic Programming, pp. 369-389, The MIT Press (1994).

[2] Reynolds, C.W.: Evolution of Obstacle Avoidance Behavior: Using Noise to Promote Robust Solutions. In K.E. Kinnear, Jr (ed.), Advances in Genetic Programming, pp. 221-241, The MIT Press (1994).

[3] Sato, Y., and Nagaya, S.: Evolutionary Algorithms that Generate Recurrent Neural Networks for Learning Chaos Dynamics. Proc. The 3rd. IEEE Intl. Conf. on Evolutionary Computation, pp. 144-149 (1996).

[4] Tamaki, H., Kita, H., and Kobayashi, S.: Multi-Objective Optimization by Genetic Algorithms: A Review. Proc. The 3rd. IEEE Intl. Conf. on Evolutionary Computation, pp. 517-522 (1996).

[5] Beyer, H-G., and Arnold, D.V.: Fitness Noise and Localization Errors of the Optimum in General Quadratic Fitness Models. Proc. The 1999 Genetic and Evolutionary Computation Conference, pp. 817-824 (1999).

[6] Parmee, I.C., and Watson, A. H.: Preliminary Airframe Design Using Co-Evolutionary Multiobjective Genetic Algorithms. Proc. The 1999 Genetic and Evolutionary Computation Conference, pp. 1657-1665 (1999).

[7] Blumel, A.L., Hughes, E.J., and White, B.A.: Fuzzy Autopilot Design Using A Multiobjective Evolutionary Algorithm. Proc. The 2000 Congress on Evolutionary Computation, pp. 54-61 (2000).

[8] Yoshihara, I., Aoyama, T., and Yasunaga, M.: GP-Based Modeling Method for Time Series Prediction with Parameter Optimization and Node Alternation. Proc. The 2000 Congress on Evolutionary Computation, pp. 1475-1481 (2000).

[9] Tominaga, D., Koga, N., and Okamoto, M.: Efficient Numerical Optimization Algorithm Based on Genetic Algorithm for Inverse Problem. Proc. The 2000 Genetic and Evolutionary Computation Conference, pp. 251-258 (2000).

[10] Goto R., and Sato Y.: Motion Analysis of Moving Objects with Genetic Algorithms. Proc. The 2001 Genetic and Evolutionary Computation Conference, pp.1276-1283 (2001).

[11] Goto R., and Sato Y.: Applicability of Genetic Algorithms to Motion Analysis of a Moving Object. Proc. The 2002 Congress on Evolutionary Computation, pp. 765-770 (2002).

[12] Roger R.B., and Donald D.M.: Fundamentals of Astrodynamics.Dover1971 &Wiley, and Sons. Inc.: Methods of Orbit Determination. New York1965

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Minimization of the Consumption Energy of a Manipulator Using the Genetic Algorithm for Obtaining Fourier Coefficients of the Approximation Functions

Yoshio Yokose
Dep. of Electrical Engineering
Kure National College of Technology
2-2-11 Agaminami, Kure, Hiroshima 737-8506, JAPAN
E-mail: yokose@kure-nct.ac.jp

Teruyuki Izumi
Dep. of Elecrtonic and Control Systems Engineering
Shimane University
1060 Nishi-Kawatsu, Matsue, Shimane 690-8504, JAPAN
E-mail: izumi@riko.shimane-u.ac.jp

Abstract

In the industrial world, many manipulators are used in automated factories. Many trajectories are determined by giving priority to the operation efficiency such as the operating time and controllability. And it may not be taking the consumption energy into consideration in the trajectory planning. This paper describes an application of the Genetic Algorithm (GA) to solve a two-point boundary value problem that minimizes the consumption energy of a manipulator with non-linear friction. Fourier cosine series with M-th order are used in this paper. A major topic of this paper is to discuss the number of terms M in order to increase the accuracy of the solution without falling into the local minimum. We have proposed to change M in the progress of evolution.

Keywords: Manipulator, Genetic Algorithm, two-point boundary value problem, Consumption Energy.

1 Introduction

When a manipulator is driven by the PTP control method, it is important to minimize the consumption energy for the trajectory. The optimal trajectory is obtained by solving the two-point boundary value problem with the restrictions of boundary conditions and dynamic equation of a manipulator. If the problem is linear, it is easy to obtain the optimal solution analytically. However, the dynamical equation of the multi-link manipulator is non-linear due to the inertia, centrifugal force and Coulomb friction in especial. So, sign function expressing Coulomb friction is replaced to the hyperbolic tangent function. However, when the non-linearity is strong, the solution is not only convergent but also a local optimal solution is obtained frequently [1][2]. It is necessary to develop the method to obtain a solution without solving the differential equation, and the GA that can search for a solution over a wide area is applied [3]. This method has disadvantages that need much calculation time for the large number of unknown variables.

We applied the method of obtaining the coefficient of the approximation function of the Fourier series by the GA for a non-linear two-point boundary value problem. This method has disadvantages that need much calculation time in order to obtain the fine solution. This paper proposes the solution of this problem, and describes application of the minimization of the consumption energy of a manipulator.

2 Dynamic equation and consumed energy of 2-link manipulator

The trajectory planning is considered to minimize the consumption energy of a horizontally articulated 2-link manipulator as the non-linear two-point boundary value problem. The angle vector of the manipulator is denoted by $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t)]^T$ as shown in figure 1 , and an angular velocity vector is $\dot{\boldsymbol{\theta}}(t) = [\dot{\theta}_1(t), \dot{\theta}_2(t)]^T$. When $\boldsymbol{i}(t) = [i_1(t), i_2(t)]^T$ is the current vector of DC motor and $\boldsymbol{K} = \mathrm{diag}(K_1, K_2)$ is the torque constant, the dynamic equation of a manipulator is as follows.

$$\boldsymbol{H}(\boldsymbol{\theta}(t))\ddot{\boldsymbol{\theta}}(t) + \boldsymbol{C}(\boldsymbol{\theta}(t), \dot{\boldsymbol{\theta}}(t)) + \boldsymbol{F_c}(\dot{\boldsymbol{\theta}}(t)) = \boldsymbol{K}\boldsymbol{i}(t) \quad (1)$$

where $\boldsymbol{H}(\boldsymbol{\theta}(t))$ is the inertia matrix as

$$\boldsymbol{H}(\boldsymbol{\theta}(t)) = \begin{bmatrix} h_{11} + h_0 \cos\theta_2 & h_{12} + h_0 \cos\theta_2/2 \\ h_{21} + h_0 \cos\theta_2/2 & h_{22} \end{bmatrix}, \quad (2)$$

$\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ is the centrifugal and Coriolis's force vector as

$$\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = h_0 \sin\theta_2 \begin{bmatrix} -(\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2/2) \\ \dot{\theta}_1^2/2 \end{bmatrix}. \quad (3)$$

where $h_{11} = I_1 + I_2 + (m_1 l_1^2 + m_2 l_2^2)/4 + m_2 l_1^2$, $h_{12} = h_{21} = h_{22} = I_2 + m_2 l_2^2/4$, $h_0 = m_2 l_1 l_2$, m_i are link masses, l_i are link lengths. These functions have non-linearity, but are continuous. However Coulomb friction $\boldsymbol{F_c}$

$$\boldsymbol{F_c}(\dot{\boldsymbol{\theta}}) = [\, F_{c1}\ \mathrm{Sign}(\dot{\theta}_1),\ \ F_{c2}\ \mathrm{Sign}(\dot{\theta}_2)\,]^T \quad (4)$$

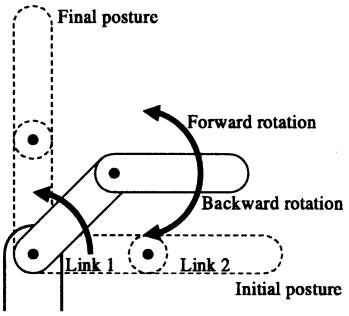


Figure 1: 2-Link Manipulator

is not continuous at the change of the sign of angular velocity. It is assumed that a manipulator moves from a start point at time $t = 0$ to a final point at time t_f. And the manipulator is controlled by PTP motion at the start time and the final time. In this case, boundary conditions are as follows.

$$\boldsymbol{\theta}(0) = \boldsymbol{\theta_0} \qquad \boldsymbol{\theta}(t_f) = \boldsymbol{\theta_f} \tag{5}$$

$$\dot{\boldsymbol{\theta}}(0) = \mathbf{0} \qquad \dot{\boldsymbol{\theta}}(t_f) = \mathbf{0} \tag{6}$$

The DC motors generate electric power in deceleration. If the power is reused effectively for the next acceleration, the consumed energy of a manipulator is Joule heat loss and the frictional heat loss as shown by

$$J = \int_0^{t_f} \{\boldsymbol{i}^T(t)\boldsymbol{R}\,\boldsymbol{i}(t) + \boldsymbol{F_c}(\dot{\boldsymbol{\theta}}(t))\dot{\boldsymbol{\theta}}(t)\}\,dt \tag{7}$$

where $\boldsymbol{R}$=diag(R_1, R_2) is resistance of the DC motor. This criterion is minimized by using the GA.

3 Approximation by Fourier series

Fourier series is applied to approximate an angle function $\boldsymbol{\theta}(t)$. Fourier series with the period of 2π is continuous. Therefore, if the initial angle differs from the final angle, the approximation function by Fourier series cannot satisfy the boundary conditions of the angles. Then, Fourier series of a half-period π is applied. The Fourier series of the half-period in the time interval $[0, t_f]$ is denoted generally by

$$\boldsymbol{\theta}(t) = \sum_{n=0}^{M} \left(\boldsymbol{a_n} \cos\frac{n\pi}{t_f}t + \boldsymbol{b_n} \sin\frac{n\pi}{t_f}t \right) \tag{8}$$

where $\boldsymbol{a_n} = [a_{1n}, a_{2n}]^T$, $\boldsymbol{b_n} = [b_{1n}, b_{2n}]^T$ are Fourier coefficients. In this paper, however, since the initial and the final velocity are 0 as shown in Eq.(6), the coefficient of the sine terms in the series should be 0. Therefore, this paper is applied to the Fourier cosine series with a half-period as

$$\boldsymbol{\theta}(t) = \sum_{n=0}^{M} \boldsymbol{a_n} \cos\frac{n\pi}{t_f}t \tag{9}$$

Two coefficients are determined from boundary conditions of Eq.(5) as follows.

$$\boldsymbol{a_{M-1}} = \frac{\boldsymbol{\theta_0} - (-1)^M\boldsymbol{\theta_f} - \sum_{n=0}^{M-2} \boldsymbol{a_n}\{1-(-1)^{n-M}\}}{2} \tag{10}$$

$$\boldsymbol{a_M} = \frac{\boldsymbol{\theta_0} + (-1)^M\boldsymbol{\theta_f} - \sum_{n=0}^{M-2} \boldsymbol{a_n}\{1+(-1)^{n-M}\}}{2} \tag{11}$$

The other coefficients are obtained by the GA. All coefficients $\boldsymbol{a_0}, \cdots, \boldsymbol{a_{M-1}}, \boldsymbol{a_M}$ are determined by these methods, and the angle functions can be approximated.

4 Application of the GA

The GA is the method for searching discrete solution space by genes. However, the trajectory function is a continuous function with respect to time. The continuous function is determined as the gene's phenotype. For example, the solution space divides to discrete values with the finite digit, when we obtain the expansion coefficients of the series by the GA. As consumption energy of a manipulator is obtained, the angle function determines by the GA. And the angle velocity function and the angle acceleration are also calculated by its differentiation. The consumption energy of Eq. 7 can be calculated from the currents $\boldsymbol{i}(t)$ of Eq. 1 . Therefore, the objective function of the GA is set to Eq. 7, and the GA searches the solution to minimize Eq. 7. The GA is applied to 2 unknown variables ($M = 3$) and 4 unknown variables ($M = 5$) for a link. If $M = 3$, lower order of coefficients $\boldsymbol{a_0}, \boldsymbol{a_1}$ are obtained by the GA, and higher order of coefficients $\boldsymbol{a_2}, \boldsymbol{a_3}$ are calculated from Eqs. (10), (11). If $M = 5$, $\boldsymbol{a_0}, \cdots, \boldsymbol{a_3}$ are obtained by the GA, $\boldsymbol{a_4}, \boldsymbol{a_5}$ are obtained similarly. We compare the result of proposed method and paper [1] in the condition that $\boldsymbol{\theta_0} = \mathbf{0}$, $\boldsymbol{\theta_f} = [1.5, 0.0]^T$ [rad], $\dot{\boldsymbol{\theta}}_0 = \mathbf{0}$, $\dot{\boldsymbol{\theta}}_f = \mathbf{0}$, $t_0 = 0.0$, $t_f = 1.0$ [s]. The other conditions are shown in Table 1. The parameters of the GA are shown in Table 2. The results of the objective function for generations are shown in figure 2. They are average values of 50 times iterations. The thin line (F 2) shows the result of 2 unknown variables, the thick line (F 4) shows 4 unknown variables. The method of 2 unknown variables

Table 1: Parameters of a manipulator

Link	i	1	2
Moment of inertia	I_i [kgm^2]	0.43	0.244
Mass	m_i [kg]	14.25	10.0
Link length	l_i [m]	0.25	0.16
Center of gravity	l_{gi}[m]	0.125	0.08
Resistance	R_i[Ω]	1.0	1.0
Torque constant	K_i[Nm/A]	1.0	1.0

Table 2: Parameters of the GA

Number of chromosomes	100
Number of elite chromosomes	20
Gene length [bits]	10
Crossover rate [%]	50
Mutation rate [%]	2

obtains a good result. However, the method of 4 unknown variables is the way of evolution even at 3000 generation. These obtained results are too coarse to approximate the angle function $\boldsymbol{\theta}(t)$ due to a few numbers of coefficients. In order to calculate a solution with more low consumption energy, it is necessary to increase the number of the terms in the series.

5 Proposed method

If Coulomb friction is taken into consideration, the current functions become discontinuous at reversing time of rotating direction. The discontinuous current functuion makes the angle functions change radically. Therefore the angle functions cannot be expressed by approximating function with a few coefficients in Fourier series. In this case, the trajectory saving energy of a manipulator may not be found. Therefore it is necessary to increase the number of coefficients so that the angle function $\boldsymbol{\theta}(t)$ can be approximated by Fourier series sufficiently. However, the following problems arise from increasing the number of unknown coefficients by the GA.

- Increase of the computation time
- Increase of the local optimal solutions

If the number of unknown coefficients is increased, although the angle function varying radically can be approximated accurately, the computation time of the

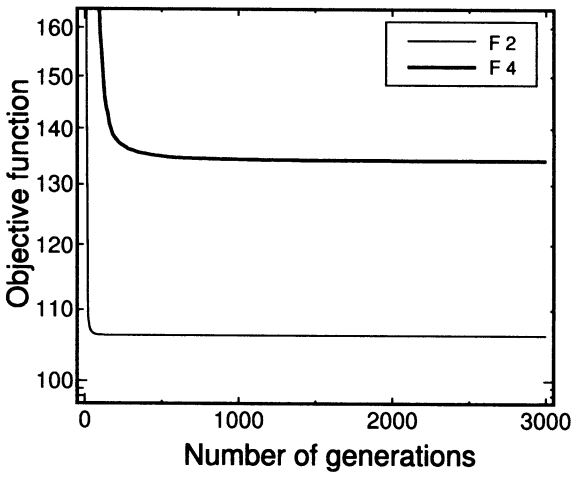


Figure 2: Transition of the objective function.

GA increases darmatically. And the local optimal solutions also increases. Therefore, when the number of unknown coefficients is increased, much time is needed for search of an optimal solution, and the probability of finding the optimal solution is lowered. So, an idea is required in order to increase the number of unknown coefficients for search by the GA.

5.1 Dynamical increment of the number of unknown coefficients

In order to find the optimal solution of the global area by the GA efficiently, the number of unknown variables should be decreased. However, in order to approximate the complicated function in accuracy, it is necessary to use many unknown variables. So, we propose the new method for increasing the number of unknown variables dynamically.

- Coarse search by a few unknown variables in the initial generations
- Increase the number of unknown variables at the later generations

In order to find the global optimal solution in a first stage of the search generations, a few design variables are set. Then it is expected that the evolution of the GA search is made be rapid, and the probability of finding a global solution is made be high. After searching coarsely due to a few unknown variables, the number of unknown variables is increased with the GA search generations.

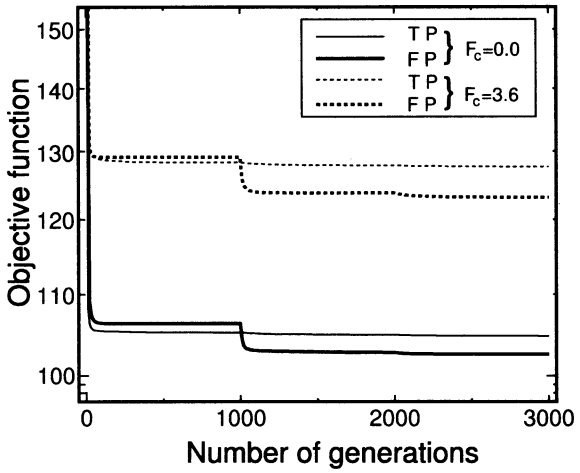


Figure 3: Transition of the objective function using the proposed method

5.2 Succession of genetic information

Increasing of the number of unknown variables is lengthening the chromosome length for search of the GA. The information on the former generation's optimal solution is used for the next generation's initial value at the change generations. The method of generating a gene for approximation function is described.

The succeeding method is applied to approximation of the angle function $\boldsymbol{\theta}(t)$ by the Fourier series. If the number of unknown coefficients in the first generation is 2, M of Eq.(9) is 3. In the first stage, if unknown coefficients are $\boldsymbol{a_0}, \boldsymbol{a_1}$ which is obtained by the GA, the calculated coefficients from Eq.(10),(11) are $\boldsymbol{a_2}, \boldsymbol{a_3}$. In the second stage, if the number of unknown coefficients is changed to 4 from 2, the coefficients obtained by the GA are $\boldsymbol{a_0}, \cdots, \boldsymbol{a_3}$, and $\boldsymbol{a_4}, \boldsymbol{a_5}$ are determined from boundary conditions. Since the good solutions $\boldsymbol{a_0}, \cdots, \boldsymbol{a_3}$ are obtained in the first stage, if they are used as one of the initial individuals in the second stage, genetic information can succeed to next generation. Similarly, genetic information $\boldsymbol{a_0}, \cdots, \boldsymbol{a_5}$ in the second stage succeed to the third stage.

6 Application of the proposed method

The parameter for GA and manipulator using in computation are the same as section 4. The proposed method is applied to the case of Coulomb friction $\boldsymbol{F_c} = \boldsymbol{0}$ and $\boldsymbol{F_c} = [3.6, 3.6]^T$ [N·m]. The result of using Fourier series for the approximation is shown in Figure 3 (FP), and the result applied to a Taylor series for comparison is shown in Figure 3 (TP), respectively. In the condition of Coulomb friction free, the

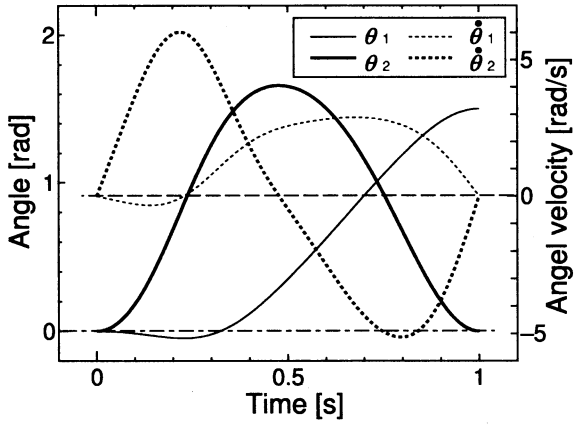


Figure 4: Optimal function of the angle $\boldsymbol{\theta}$ and the angle velocity $\dot{\boldsymbol{\theta}}$ for 2 links with $\boldsymbol{F_c} = [3.6, 3.6]^T$.

obtained values of the consumption energy are 102.6[J] (FP), 104.8[J] (TP). These values are a little larger than the consumed energy 102.3[J] of paper [1] which is obtained by solving Euler's differential equation. On the other hand, when Coulomb friction is considered, resulting consumption energy are 123.1[J] (FP) and 127.7[J] (TP). They are less than the consumed energy 133.6[J] of paper [1]. Therefore the proposed method is effective in solving two-point boundary value problem with non-linearity. Fourier series is more useful compared to Taylor series. Figure 4 shows the optimal angle function and angle velocity function approximated by Fourier series.

7 Conclusions

We proposed a new method for two-point boundary value problem. This method is applicable to the problems with the strong non-linearity that is difficult to obtain the solution by the conventional method. It is seen that the proposed method is applicable to finding the optimal solution that minimizes the consumption energy of a manipulator.

References

[1] Teruyuki Izumi, "Minimization of Energy Consumption for a Manipulator with Nonlinear Friction in PTP Motion", JRSJ, Vol. 13, No. 8, pp. 1179- 1185, 1995.

[2] Teruyuki Izumi, "Path Planning for Saving Energy of a Manipulator in PTP Motion ", JRSJ, Vol. 18, No. 7, pp. 972- 978, 2000.

[3] Yoshio Yokose, Teruyuki Izumi, "Application of Genetic Algorithms for Minimizing the Consumption Energy of a Manipulator", Proc. of the 8th Int. Symp. on Artificial Life and Robotics, Vol. 1, pp. 50- 53, 2003.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Keeping Diversity of Reference Set for Keeping Diversity of Population in Distributed Genetic Algorithms

Tomoharu Nakashima, Takanori Yoshida, and Hisao Ishibuchi
Department of Industrial Engineering
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
{nakashi, dayoshi, hisaoi}@ie.osakafu-u.ac.jp

Abstract

This paper proposes the use of reference set in genetic algorithms in order to maintain diversity of the population. Particularly, we examine the effect of the reference set on the performance of a distributed genetic algorithm. For the distributed genetic algorithm, we consider island genetic algorithms in this paper. During the execution of the algorithm, individuals with high fitness values are selected to copy in the reference set. The referene set is used to modify the fitness value of individuals. In our fitness modification, we first calculate similarity measure of an individual to its nearest reference individual in the domain space. Then, the fitness value of the individual is calculated by using the function value of the individual and the similarity measure. In computer simulations, we examine the perfromance of our proposed method on multimodal function optimization problems.

1 Introduction

Distributed genetic algorithms have been proposed and shown their high search ability [1, 2]. It is shown in [2] that distributed genetic algorithms perform better than non-distributed genetic algorithm even if the parallel implemention of algorithms is not involved. That is, the localized selection in the algorithms themselves plays an important role in the research of the genetic algorithms. The main advantage of distributed genetic algorithms is localized selection. That is, the selection is made among a small portion of a population, which makes the diversity of the population larger than that of non-distributed genetic algorithms.

It is well-known that the balance between exploration and exploitation is one of the most essential factor in any search methods. Genetic algorithms deal with this issue by considering the diversity and the selection pressure of the algorithms. That is, the diversity in the population corresponds to the exploration ability and the selection pressure corresponds to the exploitation ability of genetic algorithms. However, it is difficult to attain the best balance between the diversity and the selection pressure because if one tries to keep a large diversity in a population the search speed slows down. On the other hand, if the selection pressure is high, the population is likely to converge to a local optimal solution. This issue has been discussed in various literatures (for example, see [3]).

The performance of the distriuted genetic algorithms has been also examined. For example, Muhlenbein [7, 8] investigated their search ability for function optimization and combinatorial optimization problems.Theoretical analysis of distributed genetic algorithms have been also performed (see, for example, [6] and [5]).

In this paper, we examine the effect of fitness modification on the performance of distributed genetic algorithms. As distributed genetic algorithms, we use island genetic algorithms in this paper. The idea of our fitness modification is that those individuals that are similar to better individuals in the population should be decreased in their fitness values. We also examine the performance of the fitness modification algorithm in a dynamic function optimization problem where the shape of the function changes over generation.

2 Island GA

Island GAs are often referred to as coarse-grained genetic algorithms [4]. One of the main characteristic features of the island GAs is that the entire population is divided into multiple subpopulations. Each subpopulation undergoes a simple GA.

The procedure of an island GA is described as follows:

[Island GA]

Step 1: Initialize the population. Subdivide the entire population into multiple subpopulations. Each subpopulation forms an island.

Step 2: For each island, do

Step 2-1: Evaluate each individual in the subpopulation.

Step 2-2: Update the subpopulation using selection, crossover, and mutation.

Step 3: Migration occurs every M iterations of Step 2. In the migration procedure, a prespecified number of individuals are selected for each island using roulette wheel selection based on the fitness value. The selected individuals are copied to randomly selected islands.

Step 4: If some prespecified condition is satisfied, stop the procedure. Otherwise go to Step 2.

3 Reference Set for Diversity Keeping

In this paper, we propose the use of reference set in genetic algorithm. The objective of the reference set is to keep as much diversity as possible while efficiently searching for optimal solutions. In our fitness modification algorithm, individuals with high function values are selected from the population. The selected individuals are copied to a secondary population, which is called a reference set. The idea of our fitness modification is that individuals are refrained from converging around better individuals with high fitness values. Thus, we reduce fitness values of individuals that are near reference individuals in the domain space. Each reference individual in the reference set has a radius. Let us assume that N_{ref} individuals are selected from the population as reference individuals. The radius $Radius_j$ of the j-th reference individual in the reference set is determined as follows:

$$Radius_j = a \cdot \min_{k \neq j}\{d_{jk} | k = 1, \ldots, N_{\text{ref}}\}, \quad j = 1, \ldots, N_{\text{ref}}, \quad (1)$$

where a is a positive constant and d_{jk} is an Euclidean distance between the j-th and the k-th reference individuals in the domain space.

The idea of our fitness modification is that individuals in crowded areas should be punished by reducing the fitness value. We use the following fitness function for a function maximization problem:

$$fitness_i = f_i \times \frac{d_{ij*}}{Radius_{j*}}, i = 1, \ldots, N_{pop}, \quad (2)$$

where $fitness_i$ is the fitness value of the i-th individual, f_i is the function value, and $Radius_{j*}$ is the radius of the nearest reference individual $j*$ to the i-th individual. That is, $j*$ is determined by the following equation:

$$d_{ij*} = \min\{d_{ij} | j = 1, \ldots, N_{\text{ref}}\}, \quad (3)$$

where N_{ref} is the size of the reference set. Note that the function value is used as the fitness value without any modification if the distance d_{ij*} is larger than $Radius_{j*}$.

The fitness modification algorithm in this paper is summarized as follows:

[Fitness modification algorithm I]

Step 1: Calculate function values of individuals in the population.

Step 2: Select the best N_{ref} individuals from the population as reference individuals. Copy them in a reference set.

Step 3: Determine the radius of each of the reference individual by (1).

Step 4: For each individual i, $i = 1, 2, \ldots, N_{\text{pop}}$, do the following procedure.

(a) Find the nearest reference individual $j*$ to individual i by (3).

(b) The fitness is calculated by (2).

In this fitness modification algorithms, the whole reference set is replaced with a new set that is generated from the next population. We also consider another fitness modification algorithm. In the second modification algorithm, only a part of the reference set is updated. In the beginning of the genetic algorithm, all the reference individuals are individuals in the initial population that have best function values. From the next generation, only a few individuals whose performance is best in the population are selected and added to the reference set. Finally the extra number of individuals are selected to remove from the reference set. The selection of removed individuals is performed according to diversity measure in the reference set. That is, those individuals that are near to other reference individuals are selected as the candidate for the removed individuals. The procedure of the second fitness modification algorithm is written as follows:

[Fitness modification algorithm II]

Step 1: Calculate function values of individuals in the population.

Step 2: If the current population is an initial one, select the best N_{ref} individuals from the population as reference individuals. Copy them in a reference set. Otherwise, select the best N_{update} individuals from the current population and add them to the reference set.

Step 3: If the current population is not an initial one, do the following:

(a) For each reference indiviuals, calculate the sum of distance to other reference individuals.

(b) Remove N_{update} reference individuals that have smallest sum of the distance that is calculated in the last substep. This procedure generates a reference set with the size of N_{ref}.

Step 4: Determine the radius of each of the reference individual by (1).

Step 5: For each individual i, $i = 1, 2, \ldots, N_{\mathrm{pop}}$, do the following procedure.

(a) Find the nearest reference individual $j*$ to individual i by (3).

(b) The fitness is calculated by (2).

4 Computer Simulations

In this section, we examine the effect of our fitness modification in the last section on the performance of the genetic algorithms. We use the following functions for the function optimization problems. All of them are to be minimized.

$$\mathrm{F1} : f_1(x_i | 1 \leq i \leq 10) = 200 + \left[\sum_{i=1}^{10} x_i^2 - 10\cos(2\pi x_i) \right]$$

$$x_i \in [-5.12, 5.11] \qquad (4)$$

$$\mathrm{F2} : f_2(x_i | 1 \leq i \leq 10) = 10 - \sum_{i=1}^{10} \sin^2(\pi x_i),$$

$$x_i \in [-1.28, 1.27] \qquad (5)$$

The above functions are both multimodal functions. The function $F1$ has a global optimal and many local optima, and the function $F2$ has many global optima.

We implemented our fitness modification algorithm by using island genetic algorithms as follows: original island GA with no fitness modification (IGA), Island GA with fitness modification I (IGA+), and Island GA with fitness modification II (IGA++). Note that the fitness modification algorithm is performed only for the reference individuals from other island. That is, the reference individuals from the same island are ignored in the fitness modification algorithms.

Table 1: Basic parameter sets in the computer simulations.

Population size N	400
The size of reference set N_{ref}	8
The number of islands for IGA	4
Migration interval M for IGA	50
Crossover probability	1.0
Mutation probability	0.05
a in (1)	2.0
# of generations	500

The parameter specifications in our experiments are shown in Table 1.

We iterated each algorirhm for ten times for each function optimization problem. During the execution of the algorithms, we monitored the diversity of the population as well as the function values of the elite individuals (that is, the best individual in terms of fitness value). The diversity of the population is measured in terms of the value of variables in the solution space (variance of the real input vector for functions).

We show in Figure 1 and Figure 2 the fitness value of the best individual in the population and the variance of the population, respectively. From these figures, we can see that our fitness modification algorithm can keep more diversity maintained in the population than without it. We can also see that almost the same search ability is observed for both with and without the fitness modification algorithm.

Next we show the simulation results for $F2$ in Figure 3 and Figure 4. From these figures, we can see that the diversity of the population is higher when the fitness modification algorithms are used. The diversity of the poupulation is the highest when the fitness modification algorithm II is used while maintaining the same search ability.

5 Conclusions

In this paper, we proposed fitness modifictaion algorithms that keep diversity of the population in genetic algorithms. The fitness modification algorithm degrade the fitness value of those individuals that are similar to reference individuals. Reference individuals are selected from the population according to function value. In our computer simulations, we showed that modified genetic algorithms with our fitness modification algorithm can keep more diversity in the population than without it.

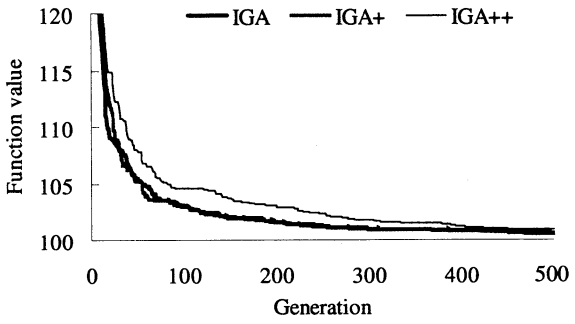


Figure 1: Fitness value for $F1$.

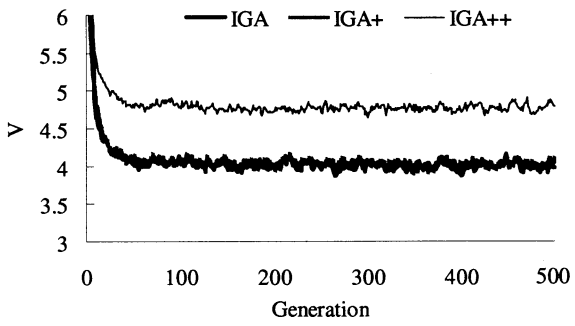


Figure 2: Variance in the population for $F1$.

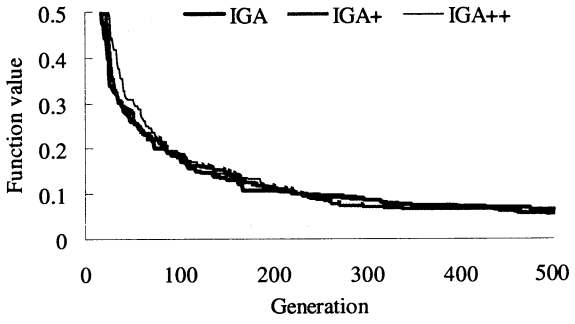


Figure 3: Fitness value for $F2$.

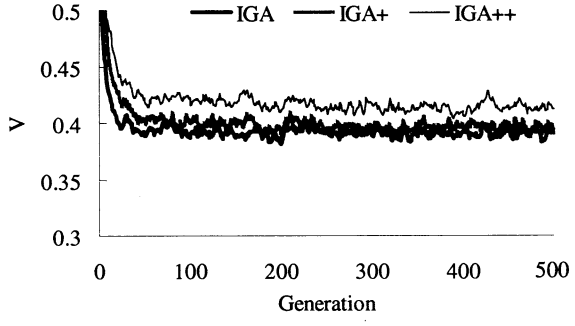


Figure 4: Variance in the population for $F2$.

References

[1] R. Tanese, "Distributed Genetic Algorithms", *Proc. of the third International Conference on Genetic Algorithms*, pp. 434–439, 1989.

[2] V. S. Gordon and D. Whitley, "Serial and Parallel Genetic Algorithms as Function Optimizers", *Proc. of the International Conference on Genetic Algorithms*, pp. 177–183, 1993.

[3] D. Whitley, "The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best", *Proc. of the third International Conference on Genetic Algorithms*, pp. 116–121, 1989.

[4] E. Cantú-Paz, *Efficient and Accurate Parallel Genetic Algorithms*, Kluwer Academic Publishers, Boston, 2000.

[5] C. C. Pettey and M. R. Leuze, "A Theoretical Investigation of a Parallel Genetic Algorithm", *Proc. of the International Conference on Genetic Algorithms*, pp. 398–405, 1989.

[6] H. -M. Voigt, J. Born, "Modelling and Simulation of Distributed Evolutionary Search Processes for Function Optimization", *Parallel Problem Solving from Nature*, pp.373–380, 1990.

[7] H. Mülenbein, "Parallel Genetic Algorithms, Population Genetics and Combinatorial Optimization", *Proc. of the third International Conference on Genetic Algorithms*, pp. 416–421, 1989.

[8] H. Mühlenbein, M. Schomisch, J. Born, "The Parallel Genetic Algorithm as Function Optimizer", *Proc. of the fourth International Conference on Genetic Algorithms*, pp. 271–278, 1991.

[9] J. Branke, "Memory Enhanced Evolutionary Algorithms for Changing Optimization Problems", *Proc. of Congress on Evolutionary Computation*, pp. 1875-1882, 1999.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Matching interpretation by utterance interaction on cooperative robot tasks

Tetsuro Kimura
Dept. of Applied Math. and Informatics
Ryukoku University
Otsu, Shiga 520-2194, JAPAN

Akira Sano
Dept. of Applied Math. and Informatics
Ryukoku University
Otsu, Shiga 520-2194, JAPAN

Abstract

We propose a framework for communication between autonomous robots using utterance information. It provides the way to decide meaning and effect of the utterances through the interaction between the robots. This framework consists of three factors, utterance, action, and interpretation. Then, the relation between utterance and interpretation of an autonomous robot is gradually modified by reinforcement learning based on reward for the selected action on an environment. In this article, this framework is applied to an mimic task between two autonomous robots.

1 Introduction

Utterance is very important and efficiency as a way to facilitate mutual understanding. It is easy to make communication a success, if we are using a common language. However, there is much language in the world, e.g. Japanese, English, Chinese and so on.

When we meet a person whose spoken language is different from our language, it can take a lot to communicate each other. On such an occasion, we can try to read his/her mind through the process of trail and error. It is this kind of trial-and-error solution that our general procedure to solve when we will confront with unknown factors. However, before everything, we have to assume his/her mind to make confirm it with the trail-and-error solution. We will draw an assumption based on the relation between utterance and mind of ourselves, namely our interpretation of utterance. This process to facilitate mutual understanding is similar to "Davidson's radical interpretation" [1]. We propose a framework for communication between autonomous robots based on our mutual understanding.

2 Framework for Communication

Our framework for autonomous robot communication consists of the abstract communication model and the reinforcement learning process.

2.1 Communication Model

The abstract communication model has three elements for each robot. These are utterance, action and internal states. In addition, the relation between the elements are defined as following.

$$i_o^* = I_s(u_o) \quad (1)$$
$$u_s = U_s(i_s, i_o^*) \quad (2)$$
$$\hat{i_s} = I_s(u_s) \quad (3)$$
$$b_s = A_s(\hat{i_s}) \quad (4)$$

where, u_o indicates utterance of other robot, and i_o^* is guessed internal state of other robot. An internal state of other robot i_o^* is guessed from an utterance u_o by interpretation system I_s of itself (eq. (1)). The robot itself can observe only the value of left part of eqs.(1)–(4).

i_s is true internal state of itself, and u_s is own utterance of itself. Then, the combination of i_s and i_o^* can be regarded as an implicit information about the environment. Hence, the robot itself adopts an utterance u_s based on an environmental situation by utterance system U_s of itself (eq. (2)).

$\hat{i_s}$ indicates guessed internal state of the robot itself which is interpreted from own utterance u_s by interpretation system I_s of itself (eq.(3)). Namely, even the own true internal state can not be observed by the

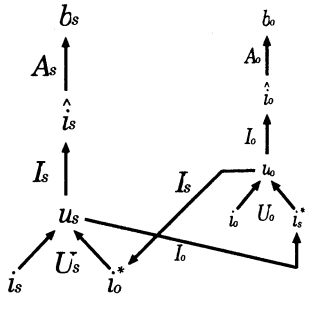


Figure 1: Relation for communication model

robot itself, and both of own internal state $\hat{i_s}$ and the other's internal state i_o^* are guessed by the identical interpretation system I_s.

b_s indicates own selected action(behavior) of the robot itself. A selected action is generated from the $\hat{i_s}$ by the action system A_s of itself (eq.(4)).

Figure 1 shows the relation among eqs. (1)–(4) with an other robot.

2.2 Reinforcement Larning

We select reinforcement learning in learning algorithms by this research. The reinforcement learning is the learning that it can adapt to situation through trial and error. There is no right answer in reinforcement learning. Autonomous robot gets relative answer called the reward while robot repeats trial and error. Q-learning is one of typical method of reinforcement learning[2]. Q-learning is algorithm that a agent select action with the large value of Q-value.

3 Experiment of Robot Task

3.1 Mimic Task

We set it up that the robots solve the tasks in each closed space.(Figure 2) This map is used simulation of robots. The robot has individual task space. The task space is space in triangle on map.

The closed spaces are completely same forms. The closed space is adjusted like different distance sensor pattern in all position. Each robot learns in individual environment. The robots know the lag of a position with each other. The action is selected from the distance between each robot and guessed internal state of itself. The robot can pinpoint his position by knowing distance sensor values and relative distance of each other. The robots learn action so that the relative distance is shorted for the task. The purpose of this task is to obtain imitated action for the robots. The existence of communication is most important for this task.

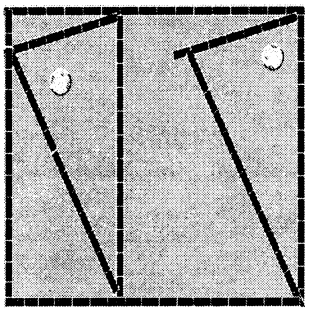


Figure 2: An experimental environment for the mimic task by two autonomous robots.

3.2 Definition of Elemental Values

The robot has 8 distance sensors. A sensor takes the value of the range 0 to 1023. They are two of front and rear, right and left. They are averaged at each direction, in short, a robot has 4 average-distance sensors, and the value of a sensor is divided into six pieces. They made into the robot's internal state, that is, a robot have the internal state pattern of $6\times6\times6\times6$. The language is made from internal of itself and internal of other. They are used each internal state which divided into 3 pieces. A robot has 81-sensor pattern for words, another robot has 81-sensor pattern for words too. The utterance is the kind of 81×81 for a robot. The interpretation guesses the internal state of self and other from sensor pattern of $6\times6\times6\times6$ through each utterance. The actions are made from 6 kinds of action, moving forward, moving back, moving diagonal right, moving diagonal left, turning left and turning right.

The learning is to get optimal action for internal state. The optimal action is to imitate the other. Rewards are important factor for the action system and the interpretation system to learn by reinforcement learning. The rewards are decided from the result of robot's action. A robot is given two kinds of reward to interact on an environment. First is given to avoid the obstacle.

Second is given to short relative distance each other.

4 Summary

We have proposed a framework to communication between autonomous robots according to utterance information. The communication mimic task has driven by the interpretation systems of each robots which is learned by the reinforcement learning through the communication task. We'll show the facilitation of utterance communication by the two-sided interpretation system.

References

[1] Donald Davidson, *Truth and Interpretation*, Oxford Univ. Press, 2001.

[2] R. S. Sutton, A. G. Barto, *Reinforcement Learning*, MIT Press, 1998.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A subspace-based face detection method using image-size reduction and magnification

Masaki Kobayashi
Dept. of Applied Math. and Informatics
Ryukoku University
Otsu, Shiga, 520-2194, JAPAN

Takashi Takahashi
Dept. of Applied Math. and Informatics
Ryukoku University
Otsu, Shiga, 520-2194, JAPAN

Abstract

We propose a new face detection method based on the subspace method. In this method, two processes of the conventional subspace method, dimensionality reduction and reconstruction are replaced with image-size reduction and magnification. The reduction process uses a conventional algorithm, and the magnification process uses the least-squared method. We apply this method to a face/non-face discrimination tasks. As a result, it obtained better classification performances than those obtained by conventional subspace method.

1 Introduction

Human face is one of the key devices for human communication, and then it is important for researches on human-robot interaction to develop techniques for detecting face regions in an image. From this viewpoint, it is necessary to investigate a face detection method considering computational costs as well as accuracy of detection. In this paper, we propose an efficient face detection method for discriminating face images from other non-face images.

In pattern recognition literature, dimensionality reduction techniques such as Principal Component Analysis (PCA) are often used to reduce the dimension of image data. The subspace method[1] is one of the typical approaches. In case of applying it to face/non-face discrimination tasks, two linear transformations are found by PCA corresponding to the two classes. Given a test data, it is classified into face or non-face class according to its reconstruction errors, that is, distances between the data and its projections by the two transformations. From the viewpoint of neural networks, these linear transformations can be regarded as three layer linear perceptrons[2] which are optimized to reconstruct face and non-face data from their dimensionality-reduced representations, respectively.

In this paper, we investigate a new method based on the subspace method. Each of the above two perceptrons is replaced with a network model composed of two computation processes, the image-size reduction process and the reconstruction process. The former reduces the size of the inputted image by a conventional interpolation algorithm, while the latter reconstructs the original image from the reduced image. Fig.1 shows changed parts between conventional method and our method.

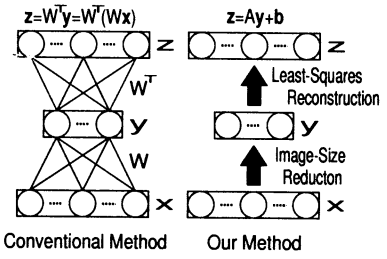


Figure 1: Conventional method and Our method

2 The subspace method for face/non-face discrimination

When applying the subspace method to face/non-face discrimination tasks, we first find two linear transformations by using PCA. Let ${}^f\mathbf{x}_i$ and ${}^n\mathbf{x}_i$ (${}^f\mathbf{x}_i, {}^n\mathbf{x}_i \in R^N$) be training data belonging to "face" class and "non-face" class, these two transformations are represented as

$$ {}^f\mathbf{y} = {}^fW\mathbf{x}, \tag{1} $$

$$ {}^n\mathbf{y} = {}^nW\mathbf{x}, \tag{2} $$

where ${}^f\mathbf{y}$ and ${}^n\mathbf{y}$ are $H(< N)$-dimensional vectors. The matrices ${}^fW$ and ${}^nW$ are defined as follows:

$$ {}^fW = [{}^f\mathbf{e}_1 \ \cdots \ {}^f\mathbf{e}_H]^T, \tag{3} $$

$$ {}^nW = [{}^n\mathbf{e}_1 \ \cdots \ {}^n\mathbf{e}_H]^T, \tag{4} $$

where ${}^f\mathbf{e}_i$ and ${}^n\mathbf{e}_i$ denote the eigenvectors of the covariance matrices of ${}^f\mathbf{x}_i$ and ${}^n\mathbf{x}_i$, respectively.

Using these transformations, an input data $\mathbf{x}$ is reconstructed as follows:

$$^{f}\mathbf{z} = {}^{f}W^{T}\,{}^{f}\mathbf{y} = {}^{f}W^{T}({}^{f}W\mathbf{x}), \quad (5)$$

$$^{n}\mathbf{z} = {}^{n}W^{T}\,{}^{n}\mathbf{y} = {}^{n}W^{T}({}^{n}W\mathbf{x}), \quad (6)$$

where $^{f}\mathbf{z}$ and $^{n}\mathbf{z}$ are N-dimensional vectors and are approximations of $\mathbf{x}$. Then, given an unlabeled data $\mathbf{x}$, it is classified by comparing two reconstruction errors $\varepsilon_f = \|^{f}\mathbf{z} - \mathbf{x}\|^2$ and $\varepsilon_n = \|^{n}\mathbf{z} - \mathbf{x}\|^2$. If $\varepsilon_f < \varepsilon_n$, the data is labeled as "face", otherwise it is labeled as "non-face".

3 Proposed method

We propose to replace the dimensionality process (Eqs.(1) and (2)) and the reconstruction process (Equations(5) and (6)) of the conventional subspace method with image-size reduction and magnification (cf.Fig1).

The image-size reduction process is repetition which reduces size of an image to 1/4.

The reconstruction process is a linear transformation magnifying the image, and its parameters are found by applying the least-squares method. Given a size-reduced image $\mathbf{y}$ corresponding to an input $\mathbf{x}$, the reconstruction process is represented as follows:

$$^{f}\mathbf{z} = {}^{f}A\mathbf{y} + {}^{f}\mathbf{b}, \quad (7)$$

$$^{n}\mathbf{z} = {}^{n}A\mathbf{y} + {}^{n}\mathbf{b}, \quad (8)$$

where A is an $N \times N$ dimensional matrix, and $\mathbf{b}$ is an N dimensional vector. Unknown parameters $^{f}A, {}^{f}\mathbf{b}, {}^{n}A, {}^{n}\mathbf{b}$ are determined by the least-squares method to minimize the error between the reconstructed image and its original version. ^{f}A and $^{f}\mathbf{b}$ are found by minimizing

$$\sum_{i=1}^{M_f} \|\mathbf{x} - {}^{f}\mathbf{z}_i\|^2 \quad (9)$$

and ^{n}A and $^{n}\mathbf{b}$ are found by minimizing

$$\sum_{i=1}^{M_n} \|\mathbf{x} - {}^{n}\mathbf{z}_i\|^2, \quad (10)$$

where M_f and M_n are the numbers of the training data of each class. The optimal ^{f}A and $^{f}\mathbf{b}$ becomes

$$^{f}A = ({}^{f}Y\,{}^{f}Y^{T})^{-1}\,{}^{f}Y\,{}^{f}X^{T}, \quad (11)$$

$$^{f}\mathbf{b} = {}^{f}\overline{\mathbf{x}} - {}^{f}A\,{}^{f}\overline{\mathbf{y}}, \quad (12)$$

where $^{f}X = [\mathbf{x}_i - {}^{f}\overline{\mathbf{x}}| \cdots \mathbf{x}_{M_f} - {}^{f}\overline{\mathbf{x}}]$, $^{f}Y = [\mathbf{y}_i - {}^{f}\overline{\mathbf{y}}| \cdots \mathbf{y}_{M_f} - {}^{f}\overline{\mathbf{y}}]$, and $^{f}\overline{\mathbf{x}}$ and $^{f}\overline{\mathbf{y}}$ are the means of $^{f}\mathbf{x}$ and $^{f}\mathbf{y}$. ^{f}A and $^{f}\mathbf{b}$ are also found alike. Classification process is in the same way as the conventional subspace method.

	Subspace method			Proposed method		
H	Face	NF	Total	Face	NF	Total
1	92.76	94.40	94.15	91.83	96.48	95.78
4	94.14	95.74	95.50	92.14	98.90	97.88
16	98.61	98.46	98.48	97.38	99.56	99.23
64	97.07	98.52	98.30	90.91	99.86	98.51
256	92.91	98.76	97.88	44.22	99.97	91.42

Table 1: The discrimination rate

4 Results

In order to evaluate the classification performance of our method, we have done some experiments on a face detection task. We prepared 735 frontal face images and 3640 non-face images as the training data to determine the parameters of the reconstruction processes. Each image was consisted of 32×32 gray pixels, so that the input vector was composed of 1024-dimension. These images were reduced to 16×16, 8×8, 4×4,⋯,etc, pixels.

They were inputted to two reconstruction networks, and then each image was labeled as face or non-face by comparing the reconstruction errors of two networks. For comparison, we have also applied the subspace method to the same data. Table 1 shows the results.

On both methods, the best classification performances were obtained in the case that the dimension of the input was reduced to 4×4 = 16. It is also shown that our method obtained better results than the subspace method.

5 Summary

In this paper, we proposed to replace the dimensionality process and the reconstruction process of the conventional subspace method with image-size reduction and magnification. As a result, it was shown that our method is more efficient for face/non-face discrimination method than the conventional subspace method. Our method will be applicable in the field of pattern recognition.

References

[1] Oja, *Subspace Methods of Pattern Recognition*, Research Studies Press Ltd., 1983.

[2] K. I. Diamantaras, S. Y. Kung, *Principal Component Neural Networks*, Wiley Interscience, 1996.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

On the Stability of a Model for Interaction between HIV and T–Cell $CD4^+$

Hiroshi UEDA and Yoshiteru ISHIDA
Department of Knowledge-based Information Engineering
Toyohashi University of Technology
Hibarigaoka 1–1, Tempaku, Toyohashi, Aichi 441–8580

Abstract

We make observational study about 4-dimensional Nowak–May's mathematical model. In this model, number of HIV strain is 2, the fraction of v_2/v_1 has effects on stability of the model. We derive theorem that can recognize the case of limit of the value of v_2/v_1.

Keywords HIV, Immune system, Dynamical system, Antigenic diversity threshold

1 Introduction

The complicated interaction has happened between the immune system and HIV in the period after infection to symptoms of AIDS [1].

Nowak, May and others described this interaction by the system of differential equations, and proposed the diversity threshold theory [2, 3].

This theory make a point that there is a conditions for increasing population of HIV.

We discuss about the Nowak–May model in the case of four dimensions.

2 4-dimensional model

In this case, the system of differential equations has two HIV strains and two CD4 T-Cells as follows:

$$\dot{v_1} = v_1(b_1 - p_1x_1) \quad (1)$$
$$\dot{v_2} = v_2(b_2 - p_2x_2) \quad (2)$$
$$\dot{x_1} = k_1x_1 - u_1Vx_1 \quad (3)$$
$$\dot{x_2} = k_2x_2 - u_2Vx_2 \quad (4)$$

where v_i and x_i are time dependent variables indicating the population size of HIV strain and that of its corresponding clones of T helper lymphocytes stimulated by HIV type i, respectively($i = 1, 2$).

The following parameters are all positive;

- b_i: the net replicating rate of HIV,
- p_i: the rate of elimination of HIV by immune response specific to type i,
- k_i: the rate of stimulation of i-specific T-cells,
- u_i: the rate of depletion of T-cells by HIV.

3 Stability of the model

Definition 1. *Stability on Nowak–May model: For any positive* $v_i(0)(i = 1, 2, \cdots, N)$,

$$lim_{t\to\infty}v_i(t) \to 0$$

Theory 1. *The stability condition for the HIV population size can be formulated as follows:*

$$\sum_{i=1}^{N} \frac{b_iu_i}{p_ik_i} < 1. \quad (5)$$

Thus, stability can be characterized by the *antigenic diversity threshold* (left–hand side of (5))

The point of our observation is the fraction of HIV populations. In the case of N=2, we determine convergent value of the fraction v_2/v_1.

Theory 2. *Let the parameters to satisfy following condition , then* v_2/v_1 *converges to positive static value* α^*

$$\frac{|b_2 - b_1|}{\min\left(\frac{p_1k_1}{u_1}, \frac{p_2k_2}{u_2}\right)} < 1 \quad (6)$$

Consequently, We can determine not only stability of Nowak–May model but also converge value of v_2/v_1.

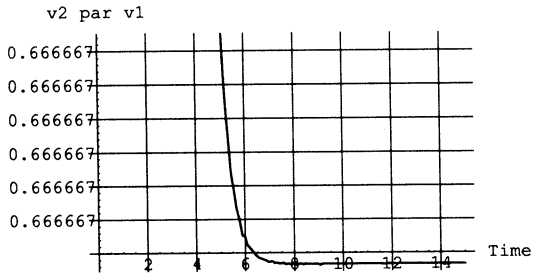


Figure 1: Time evolution of the ratio v_2/v_1 in the case of $v_2/v_1 \to \alpha^*$ and unstable.

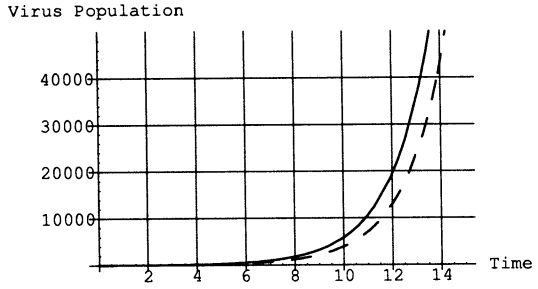


Figure 2: Time evolution of the HIV population in the case of $v_2/v_1 \to \alpha^*$ and unstable.

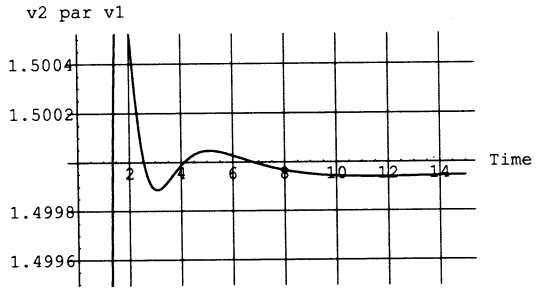


Figure 3: Time evolution of the ratio v_2/v_1 in the case of $v_2/v_1 \to \alpha^*$ and stable.

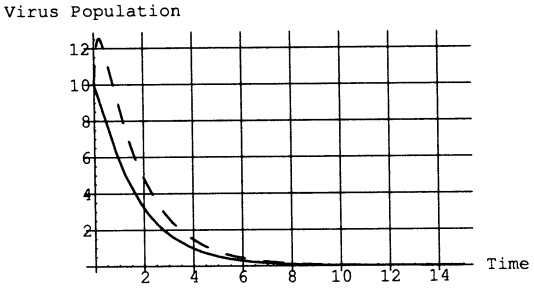


Figure 4: Time evolution of the HIV population in the case of $v_2/v_1 \to \alpha^*$ and stable.

4 Simlations

Example 1. *In the case of* $u_1 = 0.5$, $b_1 = 3$, $k_1 = 1$, $p_1 = 2$, $u_2 = u_1$, $b_2 = 3$, $k_2 = 1$, $p_2 = 3$, v_2/v_1 *converges to 0.67*$(= p_1/p_2)$ *(See figure 1). In this case, T-cells eliminate correspond HIV strain, but do not conquer to non-specific attack of HIV. As a result, the model is unstable (See figure2).*

Example 2. *In the case of* $u_1 = 0.5$, $b_1 = 1$, $k_1 = 1$, $p_1 = 2$, $u_2 = u_1$, $b_2 = 3$, $k_2 = 1$, $p_2 = 3$, v_2/v_1converges to 1.5, and model is stable (See figure 3,4). In this case, T-cells eliminate correspond HIV strain and conquar to HIV attack.

5 Discussion

The converge value of v_2/v_1 has three cases, 0, α, ∞, and only the case of v_2/v_1, the model is stable. Additionally, we can find the converge value of x_1, x_2, α^*.

However, there is unstable state in the case of v_2/v_1 converges α^*. This fact indicates variety of unstable state of Nowak–May's model as follows:

(i) In the case of unstable but $v_2/v_1 \to \alpha^*$, diversity of HIV that can be recognized by the immune system is low.

(ii) In the case of unstable but $v_2/v_1 \to \infty, 0$, diversity of HIV is high.

For the furture work, analysis of the model in the case of $N > 2$ is mentioned.

Acknowledgements

This work was supported in part by a the 21st century COE (Intelligent human sensing) from the Ministry of Education, Culture, Sports, Science and Technology.

References

[1] Martin A. Nowak and Andrew J. McMichael. How HIV defeats the immune system. *Scientific American*, pp. 58–65, August 1995.

[2] Martin A. Nowak, Roy M. Anderson, Angela R. McLean, Tom F. W. Wolfs, Jaap Goudsmit, and Robert M. May. Antigenic diversity thresholds and the development of AIDS. *Science*, Vol. 254, pp. 963–969, 1991.

[3] Rob J. de Boer and Maarten C. Boerlust. Diversity and virulence thresholds in AIDS. *Applied Mathematics*, Vol. 94, pp. 544–548, January 1993.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Towards Fully Distributed Cognitive Systems

Peter Sapaty [1] Kazuhiko Kawamura [2] Masanori Sugisaka [3] Robert Finkelstein [4]

[1] Institute of Mathematical Machines and Systems, National Academy of Sciences
Glushkova Ave 42, 03187 Kiev, Ukraine,
+380-44-2665023, +380-44-2666457(fax), sapaty@immsp.kiev.ua

[2] Center for Intelligent Systems
VU Box 350131 Sta B, Vanderbilt University, Nashville, Tennessee 37235-0131 USA
+1-615-322-2735, +1-615-322-7062 (fax), kaz.kawamura@vanderbilt.edu

[3] Department of Electrical and Electronic Engineering, Oita University,
700 Oaza Dannoharu 870-1192 Japan,
+81-97-554-7831, +81-97-554-7841 (fax), msugi@cc.oita-u.ac.jp

[4] Robotic Technology Inc.
11424 Palatine Drive, Potomac, Maryland 20854
+1-301-983-4194, +1-301-983-3921 (fax), RobertFinkelstein@compuserve.com

Abstract. A new paradigm, including a philosophy, model and technology, for the creation of advanced distributed cognitive systems is discussed. This paradigm allows for the description of interacting deliberative, reactive, and reflective processes on a semantic level. It provides new degrees of freedom for autonomous robotic teams, where the collective behavior of robots emerges as a derivative of parallel and distributed interpretation of a language, known as *WAVE-WP*, in a united physical and virtual world.

Keywords: Cognitive systems, collective behavior, multiple robots, WAVE-WP language, parallel interpretation.

1 Introduction

Using robots (and intelligent software agents) in collectives, groups, and swarms - up to the size of armies or colonies [1] - is being considered for civil and military applications. This may increase productivity and robustness in unfriendly environments, where individual robots may fail but the mission survives and fulfills objectives.

Solving complex problems by groups of robots may, however, meet serious difficulties. For example, a single human brain may be organized much better as a system than a group of brains (a *system of systems*). Thus an individual may compete successfully with a group.

Traditional control systems are usually oriented on single robots, both ideologically and by original implementation, and making intelligent groups of them may need revision of basic principles of intelligent systems.

We will discuss how to preserve the existing ideology and practice of developing advanced cognitive systems, with already developed integrity and clearness, applying them in a way suitable for covering distributed multi-robot organizations as well. The main idea is to describe their reactive, deliberative, and reflective processes on a higher, semantic level, abstracting from implementation. They may reveal potential parallelism in distributed environments, which may be automatically converted into a group behavior.

2 Cognitive systems

Cognitive systems belong to the most advanced class of intelligent systems -- the ones aware of what they are doing. While cognitive systems include reactive and deliberative processes, they also incorporate mechanisms (e.g., a *meta system*) for self-reflection and adaptive self-modification, as shown in Fig. 1 (STM and LTM staying, respectively, for short and long term memories), see also [2].

A fully cognitive robot should be able to recognize situations in which its reactive and reasoning abilities fall short of meeting task demands (i.e., a cognitive system should know what it does not know). And it should be able to make reasoned modifications to those abilities in hopes of improving the situation. A multiagent-based cognitive robot architecture with three distinctive memory structures,

namely short-term, long-term and working memories, is proposed in [3].

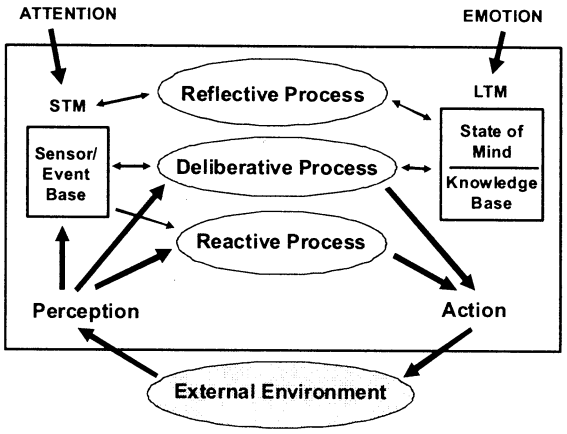


Fig. 1 Exemplary cognitive system architecture

3 Problems in developing distributed cognitive systems

Traditional intelligent systems, including cognitive ones, are oriented to robots, rather than to the tasks they should solve, just as in years past the focus of programming was on computers rather than problems. Groups of robots present a more difficult problem, such as how to integrate them into a team while preserving their individual capabilities.

All current systems and initiatives, even the most advanced like JAUS (Joint Architecture for Unmanned Systems), for example [4], have underlying agent and message-passing ideology, which, in a broader sense, inherits the analytical approach for dealing with large systems, i.e., from being defined in advance as parts to becoming the unknown whole (where the latter is rapidly becoming less observable and more vague when the number of parts and their interactions grow, and especially when systems are open and change structure over time).

A higher-level, distributed processing and control model and language, WAVE-WP, allows for the description of intelligent systems, including cognitive systems, on a semantic level and as an integral part of the environment, rather than being separated from it.

4 Distributed worlds in WAVE-WP model

One of the main features of WAVE-WP is the representation of the distributed world it operates upon, as briefly described below.

Physical world (or PW) is continuous and infinite in WAVE-WP. Existing at any of its points, and possibly performing a job, it is symbolically considered as residing in a *node*. Such a node, reflecting only occupancy (or occupancies) at the point, does not have a personal identity or content. It disappears with the termination of occupancies in it. But PW nodes have coordinates.

Virtual world (or VW) is discrete and interlinked in WAVE-WP, and, similar to the WAVE model, is represented by a Knowledge Network (KN). Its nodes may contain established concepts or facts, and links (oriented as well as non-oriented) connecting the nodes may reflect different relations between nodes. VW nodes may exist regardless of the presence or absence of occupancies or activities in them.

United PW-VW world. The same model can work simultaneously with PW and VW, where some parts are physical, while others virtual. It is also possible to have a deep integration of the two worlds, as Physical-Virtual World (or PVW, same as VPW), in which any element may have features of the both.

A very simplified example of the united PVW world is shown Fig.2.

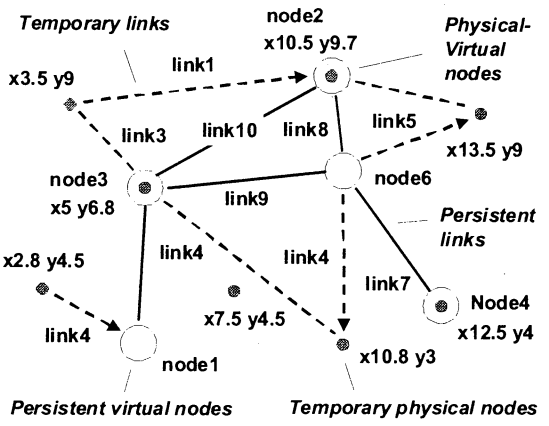


Fig. 2 United distributed physical-virtual world

A variety of effective access mechanisms to nodes, links, and their groups, say, by physical coordinates, electronic addresses, by names, by traversing links, etc. (classified as *tunnel* and *surface* navigation) abound in the model, with selective and broadcasting mode. Any representation of external and internal worlds of intelligent robotic systems can be made using this united dynamic network formalism.

5 Spatial control with waves

Solutions of any problems in this integrated world are

represented as its parallel and coordinated navigation (or exploration, invasion, grasping, coverage, flooding, conquest, etc.) by some higher-level forces, or "waves". These bring local operations, control, and transitional data directly into the needed points of the world, to perform jobs there. The obtained results, together with same or other operations may, in their turn, invade other world parts, and so on.

As already mentioned, WAVE-WP is not a message-passing or agents-based paradigm—it is a higher-level model which uses message passing and agents on implementation levels only. Its top level recursive syntax (with programs called *waves*) follows.

wave → { *advance* ; }
advance → { *move* , }
move → *constant* | *variable* |
{ *move act* } | [*rule*] (*wave*)

Braces above show zero or more repetitions of the construct, with a delimiter at the right (a semicolon, comma, and acts symbols), and square brackets identify an optional construct. A separate semicolon and comma, respectively, indicate successive and potentially parallel parts.

The successive parts, or *advances*, develop from all nodes of the Set of Nodes Reached (or SNR) of the previous advance. Parallel parts, or *moves*, all develop from each node of SNR of the previous advance, adding their SNR to the SNR of the advance.

Acts may represent local data processing, hops in both physical and virtual spaces, and local control. *Rules* establish non-local constraints and contexts over the space-evolving waves (like giving a power to create and modify the navigated worlds), and also allow the use of WAVE-WP as a conventional language. *Nodal variables* are shared by different waves in the nodes; *frontal variables* travel and replicate with waves, and *environmental variables* provide an access to different application and internal implementation infrastructures.

Arbitrary complex space-navigating, control, and data processing scenarios (data representing both information and physical matter) can be expressed in WAVE-WP in fully distributed and parallel mode. Many cooperating or competing spatial wave processes can operate in the same distributed world, communicating with multiple users, if needed.

6 Elementary programming examples

Let us consider some elementary examples of parallel operations in WAVE-WP on the distributed united world shown in Fig. 2.

Example 1. Starting in node 6, go to new PW positions x2.8 y4.5, x7.5 y4.5, and x13.5 y9, measure temperature there, and assign its highest value to variable Max in node 6:

```
direct # node6; Max = max(direct # (x2.8
y4.5, x7.5 y4.5, x13.5 y9); QUALITIES:7)
```

where physical access to the external world is performed by environmental variable `QUALITIES` of the language (using its seventh element responsible for this parameter).

Example 2. Enter all existing PW-bound nodes in the region with center coordinates x5.5 y6 and search range (radius) 4; assign `terminate` to variable `Nm44` in all found nodes (say, leaving message to the workers in these nodes to stop working); and then perform the rest of the program (wave) there:

```
direct # x5.5 y6 r4; Nm44 = terminate;
wave
```

This program will move simultaneously to locations x2.8 y4.5, x3.5 y9, x5 y6.8, and x7.5 y4.5 (see Fig.2) lying in the specified region (there is also VW-bound node3 among them). Program control will split, and the rest of wave will replicate, performing the mentioned jobs in each node.

Example 3 (related to topology processing). Starting from all nodes, find all triangles in relations (links) between nodes in Fig. 2 (guaranteeing their single copies only), sending the collected result directly to `USER`, who may be remote:

```
USER =
(direct # any; F = CONTENT; any # any;
ADDRESS < BACK; F&= CONTENT; any # any;
ADDRESS < BACK; F &= CONTENT; any # F:1;
F)
```

Frontal variable `F` (propagating in multiple copies throughout the network with splitting waves) accumulates the result, finally sent to the user. In the dense form, using allowed abbreviations of the keywords, this program may be shorter:

```
U=(@#;F=C;#;A<B;F&=C;#;A<B;F&=C;#F:1;F)
```

The user will receive the following collected result:

```
x3.5y9, node2, node3,
node2, node3, node6,
node2, node6, x13.5y9,
x10.8y3, node3, node6
```

The VW-bound nodes are represented in the result by

their names, whereas pure PW nodes, having no names or contents (due to the language semantics), are represented by their coordinates instead.

Many examples of distributed processing of complex knowledge structures in the predecessor language, WAVE, can be found in [5], with tasks solved via the Internet related to graph theory, pattern matching, knowledge base search, distributed simulation, and virtual reality.

In the parallel and distributed examples above, as well as in all other parallel and distributed projects in WAVE, on the application level there were no agents and message passing at all—these were involved on the automatic implementation layer only, making application programming hundreds of times shorter, and respectively, simpler.

7 Distributed WAVE-WP Interpreter

WAVE-WP is executed by the networks of wave interpreters (or WI) operating cooperatively and in parallel with each other, each WI comprising: Control Processor, Parser, Operation Processors, Communication Processor, Wave Queue, Track Forest, Nodal Variables, Frontal Variables, Environmental Variables, Knowledge Network, and Incoming and Outgoing Queues [5]. These modules form parallel interpreter architecture that can be easily implemented in both software and hardware.

The interpreter, if used in unmanned mobile platforms, may be integrated with special software or hardware tools and gadgets like neurocomputers (N), motion control (M), vision control (V), manipulators of physical matter or objects (PMO), specific sensors (S), as well as with those providing communication with other interpreters using radio, radar, laser or sonar channels (tactile too, for direct contacts), as shown in Fig.3, also see [6,7].

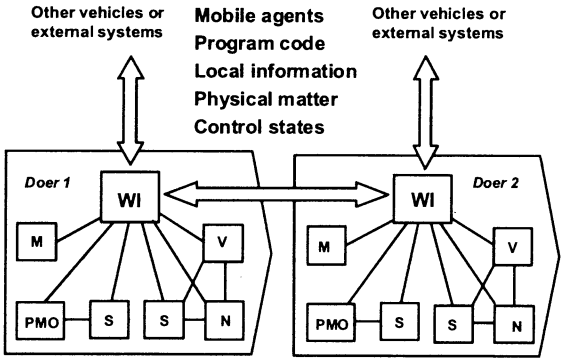


Fig. 3 Integration of WAVE-WP interpreter with other systems and tools

8 Distributed cognitive processes in open environments

A preliminary analysis and study of advanced intelligent control models and systems as, for example, described in [8], shows no limits for expressing existing knowledge processing, planning, image recognition, etc., using the spatial WAVE-WP automaton directly operating with complex network topologies. This allows for the description of complex deliberative, reactive, and reflective processes as a whole, all operating directly in distributed environment, on the same shared knowledge networks, in a highly integral and absolutely seamless mode. The result may be considered a kind of ubiquitous, spatial, body-less, global mind, which is shown in an abstract form in Fig. 4.

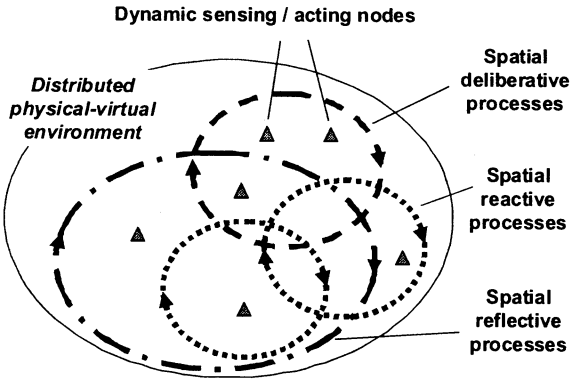


Fig. 4 Fully distributed representation of the cognitive processes

9 Automatic mapping of spatial cognition onto dynamic networks of doers

Such an enormous freedom not only from implementation hardware but from the traditional partitioning of the system into agents and their interactions, allows us to make an efficient automatic implementation by any teams of communicating doers of any (technical as well as biological) origin.

This implementation may be very robust as the peculiarities of the WAVE-WP model presume free runtime migration of any parts of the networked worlds (or their whole) and associated processes between doers. In Fig 5, a snapshot of migrating spatial cognitive processes is shown in the group of four doers (robots). Some of the processes are local to current doers, while others involve groups of them, pursuing global goals while behaving altogether as one integral, cognitive system.

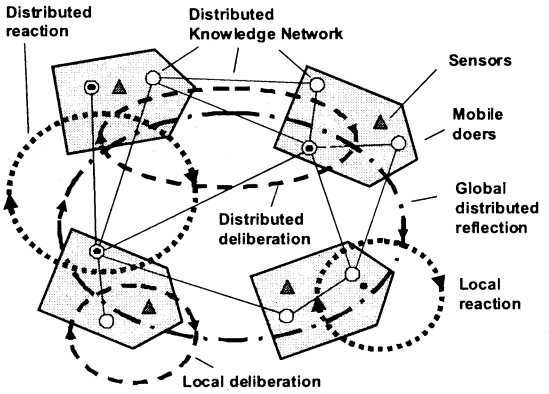


Fig. 5 Distributed mapping onto the world of doers

Using this approach, the general mission of a cognitive system and its functionality can be constantly maintained as a whole, with automatic implementation effectively delegated to a WAVE-WP interpreter which provides dynamic mapping of this whole onto the robotic hardware. The technology can involve at runtime any needed (or available) number of robots, to fulfill the mission objectives, with collective behavior of robots being a derivative of the mission parallel and distributed interpretation process.

The mission-to-hardware mapping process is fully distributed, not requiring central resources, and each robot may happen to be involved at any level of the distributed command and control process in any moment of time. Failed robots can be automatically substituted at runtime without loss of the overall mission integrity.

10 Conclusions

The proposed paradigm can keep high intelligence in the group as a whole, rather than in individual robots, which may be destroyed. It has strong advantages over traditional agent-based approaches, where each robot is designed and tasked individually at the start of a task or mission. This technology creates agents (including mobile robots) and their interactions, synchronization, and mutual subordination dynamically, on the implementation layer (often emergently), drastically reducing and simplifying the application programming of cognitive systems. More on the WAVE-WP technology and its current and previous applications can be found on the Internet, for example, in [9].

References

[1] Arkin, Ronald and George Bekey, *Robot Colonies*, Kluwer Academic Publishers, 1997.

[2] K. Kawamura, D.C. Noelle, K.A. Hambuchen, and T.E. Rogers, "A multi-agent approach to self-reflection for cognitive robots", *Proc. of 11th Int'l Conf. on Advanced Robotics*, Coimbra, Portugal, June 30 - July 3, 2003, pp. 568-575.

[3] K. Kawamura, C.A. Clifton, K.A. Hambuchen, and P. Ratanaswasd, "MultiAgent-Based Cognitive Robot Architecture and its Realization: Towards Body-Mind Integration", *IEEE International Conference on Robotics and Automation (ICRA) New Orleans, April 26 – May 1, 2004.*

[4] The JAUS Tutorial, http://www.jauswg.org/JAUStutorial.ppt.

[5] P. S. Sapaty, "Mobile Processing in Distributed and Open Environments", John Wiley & Sons, ISBN: 0471195723, New York, February 1999, 436 p.

[6] P. Sapaty, M. Sugisaka, "Towards a Universal Distributed Brain for Mobile Multi-Robot Systems", Journal of Mathematical Machines and Systems, No. 3-4, 2003, IMMSP, Kiev, 2003, pp. 3-24.

[7] M. Sugisaka, P. Sapaty , K. Imamura , K. Tokuda , N. Masuda, R. Finkelstein, J. Albus, "Research on Biologically Inspired Bipedal Dynamic Humanoid Robots", Proc. World Symposium Unmanned Systems 2003, July 15-17, 2003, Baltimore Convention Center, Baltimore, USA.

[8] J. Albus, R. Finkelstein, et al., 4D/RCS: A Reference Model Architecture For Unmanned Vehicle Systems. Version 2.0, Technical report, Intelligent System Division, NIST, 2002.

[9] P.S. Sapaty, "Over-Operability in Distributed Simulation and Control", The MSIAC's M&S Journal Online, Winter 2002 Issue, Volume 4, No. 2, Alexandria, VA, USA, http://www.msiac.dmso.mil/journal/WI03/sap42_1.html.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Evolution of biological concept network

Tetsuya Maeshiro
Sch. Lib. Info. Sci
Univ. Tsukuba
Ibaraki, 305-8550 JAPAN
and
HIS, ATR International

Katsunori Shimohara
HIS
ATR International
Kyoto, 619-0288 JAPAN

Shin-ichi Nakayama
Sch. Lib. Info. Sci
Univ. Tsukuba
Ibaraki, 305-8550 JAPAN

Abstract

Biological technical terms extracted from dictionaries and textbooks were organized according to semantic relationships. Extraction of semantic relationships is automatic, using algorithms based on formation rules of technical terms and parsing results of texts containing technical terms. Biological technical terms represent biological concepts, thus network representation, where nodes represent technical terms and links represent semantic relationships, is a network representation of biological knowledge. Biological concept network evolves as follows. First, newly generated biological concept connects to related concept in concept network. Then this concept preferably connects to other concepts with small number of links, which are concepts in the same biological knowledge field. With the advance in the biological field to which the new concept was connected, this new concept successively connects to other concepts, becoming a more general biological concept.

1 Introduction

Suppose a concept network of a scientific field, where nodes and links model respectively concepts and semantic relationships among concepts, represents knowledge of the field. Then advance in a scientific field can be treated as the evolution of concept network, as the structural change in concept network reflects the advance in the scientific field. Under such assumption, we have studied biological knowledge and have found that a possible mechanism of evolution of concept network related with the addition and deletion of concepts.

Another assumption is that technical terms represent concepts in the scientific field. Some advantages in treating technical terms are well defined meaning and clear formation rules of composite terms. Therefore, the use of technical terms is equivalent to treat concepts of the target field, and extraction of semantic relationships among concepts is possible by applying formation rules of technical terms.

Change in concept network is (i) generation of new concept; (ii) deletion of an existing concept; and (iii) change in semantic relationships among concepts.

This paper discusses the evolutionary mechanism of knowledge from the viewpoint of concept network, with changes involving generation and deletion of concepts.

2 Extraction of Terms and Semantic Relationships

Extraction of semantic relationships is automatic, using algorithms based on formation rules of technical terms and parsing results of texts containing technical terms. Extracted semantic relationships include hierarchical and equivalence relations.

Following methods are used to extract semantic relationships among concepts.

2.1 Biological Terms

133,251 biological terms were extracted from dictionaries and textbooks of Biology, Biochemistry, Molecular Biology, Neurobiology, Immunology and Microbiology, and from Medline MeSH.

2.2 SS-KWIC

SS-KWIC [1] is a method to detect hierarchical and association relations based on formation rules of technical terms. SS-KWIC decomposes a technical term into basic terms that constitute the technical term, and extracts hierarchical and association relationships among basic terms and the technical term. With the

hierarchical relationship, it is possible to use more general or more specific concepts than the target concept. Similarly, the association relationship provides related concepts.

Formation rules are language dependent, and appropriate algorithms are used each language. For English, the algorithm is as follows. Let T be a technical term, composed of n basic terms (B_1, B_2, ..., B_n) in this order, p be a preposition, and B be the basic term under consideration. The relation between T and B is determined as follows.

The relation is hierarchical if

$T = (\ldots, B, p, \ldots)$

$T = (\ldots, B)$, no preposition before B.

The relation is association if

$T = (B, \ldots)$, no preposition after B.

$T = (\ldots, B, \ldots)$, no preposition before or after B.

$T = (\ldots, p, \ldots, B, \ldots)$.

2.3 C-TRAN

C-TRAN [2] is a method to detect synonym terms from translated terms in multiple languages and sets of synonym terms in each language. The detection is possible because terms in a synonym set of a language may have translated terms belonging to different synonym sets in another language. For instance, let A_1, A_2, and A_3 be terms in language A, and B_1, B_2, and B_3 be terms in language B. Suppose (A_1–B_1), (A_2–B_2), and (A_3–B_3) be translation pairs, and A_1 be the synonym of A_2 in language A, and B_1, B_2, and B_3 constitute a synonym set in language B. Then A_1, A_2 and A_3 may constitute a synonym set, detected by the translation of B_3 to A_3. A check after the operation is necessary because translated terms is frequently not constituted purely with true synonym terms, and C-TRAN also detects hierarchical relations and false synonyms when a term has multiple meanings.

2.4 Parsing

Parser is also used to extract semantic relationships among terms, processing text data in textbooks.

3 Discussions and Conclusions

Biological technical terms represent biological concepts, thus network representation, where nodes represent technical terms and links represent semantic relationships, is a network representation of biological knowledge. The number of links per node seems to indicate the importance of the concept in Biology, as top concepts are all essential biological concepts (Table 1). These concepts function as hubs in the concept network, resulting in a decrease of shortest path length among concepts. The measured path length is 2.71.

Table 1: Top 20 terms with highest number of semantic relationships (links)

1	Cell	11	Kinase
2	Protein	12	Antibody
3	Gene	13	Peptide
4	Virus	14	Mutation
5	Disease	15	Tissue
6	Enzyme	16	Animal
7	Membrane	17	Human
8	Antigen	18	Transcription
9	Growth	19	Hormone
10	Mouse	20	Tumor

Analysis indicate the biological concept network is a small world network, with small path length and high clustering coefficient, and also presents scale free property. The scale free property, however, differs from other small world networks, and shows double scale free property, which seems to be particular to networks derived from natural language [3].

Focusing on the hierarchical structure of concept network, it is possible to visualize the concept network according to the generality of concepts, where general concepts are positioned on the top and specific concepts at bottom. As Table 1 indicates, generality of concepts is related with the number of semantic relationships. Therefore, such visualization is equivalent to position concepts according to the number of semantic relationships, i.e., to position nodes according to the number of links (node connectivity).

In order to analyze the hierarchical structure based on the node connectivity, we analyzed the ratio of the node connectivity of directly connected nodes which has smaller number of links. This analysis is closely related to the algorithm of network formation or evolution, i.e., the formation of biological knowledge from the viewpoint of concept network. A well known network evolution algorithm involving node addition that results in scale free network is the proportional selectivity algorithm, where the selection probability of the target of a newly inserted node is proportional to the number of links of the target node.

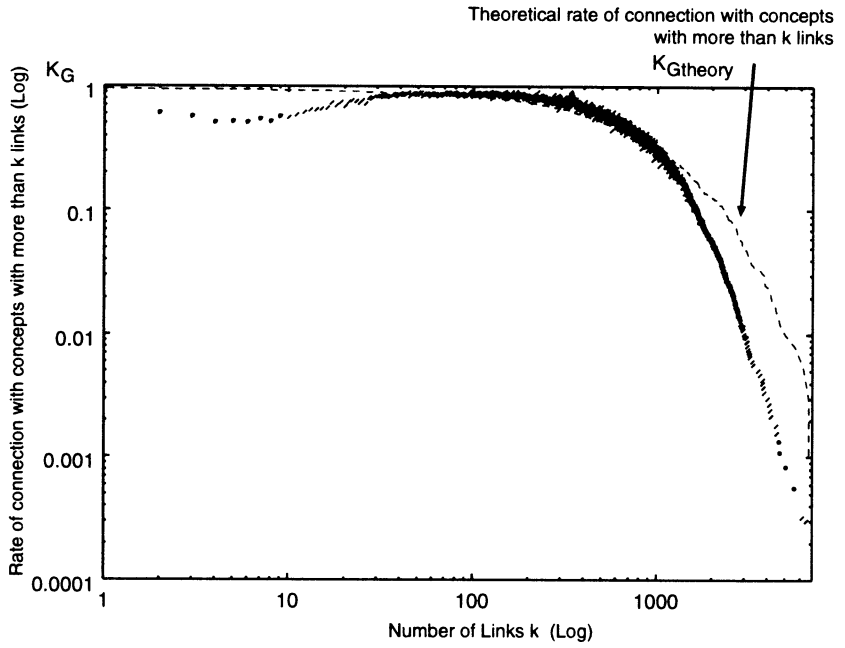


Figure 1: Rate of connection with concepts with more than k links, where k is the number of links of concept under consideration. Measured and theoretical (dotted) curves.

The theoretical value $K_{G\text{theory}}$ is given by

$$K_{G\text{theory}}(k) = 1.0 - \frac{1}{N_w}\sum_{i=1}^{k} n(i) \times i \qquad (1)$$

$$N_w = \sum_{i}^{k_{max}} n(i) \times i \qquad (2)$$

where $n(i)$ is the number of concepts with i semantic relationships (links), k_{max} is the maximum number of semantic relationships of node in the concept network, and N_w is the weighted total number of concepts.

Comparison of the theoretical curve of proportional selectivity algorithm with the curve generated from the constructed biological concept network indicates that two curves match for middle range of number of links. Smaller values of K_G for concepts with large number of links (hub concepts) suggest that hub concepts are loosely connected, each hub concept consituting an "independent" cluster of concepts. Therefore, biological concept network consists of a set of loosely coupled groups of concepts, where each concept is a basic or fundamental biological concept (some listed int Table 1). This indicates the structure of biological concept network reflects the structure of biological knowledge, as Biology consists of subfields.

However, for large number of links, the discrepancy increases for higher number of links. Another interesting result is that concepts with small number of links, which are newly introduced concepts or very specific concepts, form clusters partially separated from concepts with larger number of links. concepts with links between 4 and 10 have only 50% of links connected to concepts with more links, a value much smaller than the theoretical value (nearly 100%). Therefore, concepts with small number of links are tightly connected with each other with probability much higher than expected from probability theory, forming small clusters.

This suggests an existence of a valley that a newly introduced biological concept should surpass in order to be incorporated into biological knowledge. Concepts that did not surpass the valley is easier to extinguish, because deletion of a concept is the result of deletion of links (semantic relationships). Generation of a semantic relationship with hub concepts or "sub" hub concepts enables the crossing of valley and the concept becomes a more solid biological concept.

Therefore, biological knowledge consists of two hierarchical levels, and this threshold of number of links is different from the threshold observed in double scale free property (around 1,000 concepts.)

The analysis suggest that the biological concept

network evolves as follows. First, newly generated biological concept connects to related concept in concept network. Then this concept preferably connects to other concepts with small number of links, which are concepts in the same biological knowledge field. With the advance in the biological field to which the new concept was connected, this new concept successively connects to other concepts, becoming a more general and solid biological concept. Opposite scenario is the new concept increases no links, eventually being deleted from the biological knowledge and becoming an obsolete biological concept.

This paper presented a simple analysis based on the number of links of concepts, and a possible evolutionary mechanism of biological concept network and consequently of biological knowledge. Further analysis with more detailed concept network is necessary.

Acknowledgements

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, "Research on Human Communication".

References

[1] Y. Fujiwara and J. Lai, "An information-base system based on the self-organization of concepts represented by terms", *Terminology*, Vol.3, pp.313-334, 1997.

[2] Y. Fujiwara, W. G. Lee, Y. Ishikawa, A. Nishioka, H. Hatada and S. Fujiwara, "Dynamic thesaurus for intelligent access to research databases", *Proc. 47 FID Conf. (Helsinki)*, pp.173-181, 1988.

[3] R. Albert and A.-L. Barabasi, "Statistical mechanics of complex networks", *Reviews of Modern Physics*, Vol.74, pp.47-97, 2002.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Application of Animal Feeding Behavior Learning to a Small Mobile Robot

Naoya Sakamoto , Hideaki Kanoh
Dept of Mechanical Engineering Informatics
Meiji University
Kawasaki City, 214-8571 JAPAN

Abstract

In a supervised learning, generation of an instruction signal is very difficult for the problem in which an appropriate output is not self-evident. Reversely an unsupervised learning occasionally suffers from wondering in local minimum region. To avoid the defects of both learning processes, a hybrid learning process of combination of two kinds of learning is proposed. We use a supervised learning for the rough searching whereabouts of the bait and use an unsupervised learning for acquiring a bait. While a supervised learning is used to train the neural net in the acquisition of the primary action pattern, the unsupervised learning is carried out for the switching to the second action pattern from the primal one and acquisition action of bait. By proposed procedure, obstacle avoidance and acquisition of a bait are able to be autonomously possible on the unknown environment. We carried out simulation and experiment using a Khepera robot and its simulator.

[Key Words] Behavior Learning, KiKS, Khepera robot, Neural network, Reinforcement learning.

1 Introduction

Recently, the discussion on the artificial life is widely carried out. The research is advanced on the theme how to imitate the action observed to the organism of the nature. Using the neural network, we develop action strategy for small animals that survive by searching food while avoiding obstacles.

The reinforcement learning is the most convenient and effective way to meet with autonomous animal feeding behavior problems in those dynamic environment using an unsupervised learning[1].

The optimization of the route of searching labyrinth and of the process of attainment to a specified substance using the reinforcement learning using a neural network has been extensively studied[2][3][4][5].

In the discovering or searching unknown position of an object, however, these methods are not effective or cannot work.

In this paper, we discuss a loitering process of the searching action of a robot for unknown position of a target(bait). We use a supervised learning for the rough searching whereabouts of the bait and use an unsupervised learning for acquiring a bait. Also the unsupervised learning is used to train the switching to the acquisition action from the loitering action. We use a Khepera robot[6] and its simulator to perform these actions.

2 Mobile robot and simulator

In our research, a Khepera robot and its simulator is used in order to imitate the bait search of an animal. Figure 1 shows the sensor arrangement of the Khepera robot. The robot has eight pair of a proximity sensor

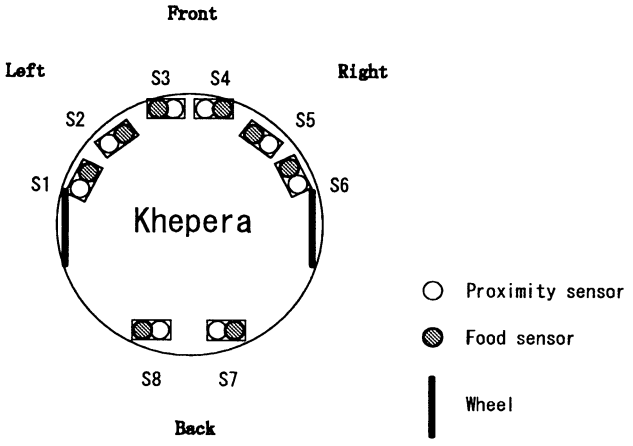


Fig.1 Sensor arrangement of Khepera robot

and a food sensor located same position. The proximity sensor returns integers values from 0 to 1023 corresponding to the distance from the obstruction and the food sensor is an infrared sensor used in a passive

mode. The robot has the motor for respectively moving two wheels. The bait is represented in this field as a light source. The proximity sensor returns large value when the obstruction is near and returns small value for the far obstruction. While the infra red sensor returns smaller value as the robot approaches to the light source as shown in Figure 2. The robot is

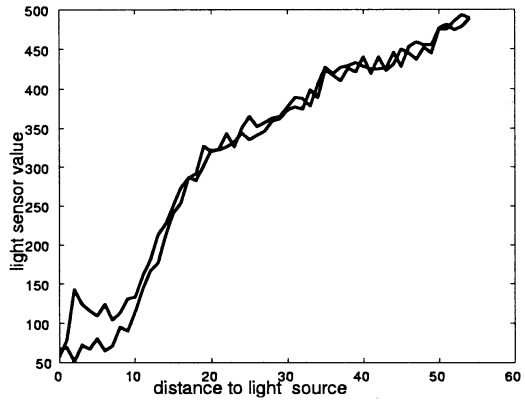


Fig.2 Light sensor's output value

connected with the computer by the serial(RC232C) cable. We use a simulator KiKS[7] for simulation of the behavior of the robot. KiKS operating on the Matlab is designed as a simulator of Khepera and is able to manipulate the Khera robot. The experiment environment is in the square of 1000 × 1000cm environment, and baits are located at random.

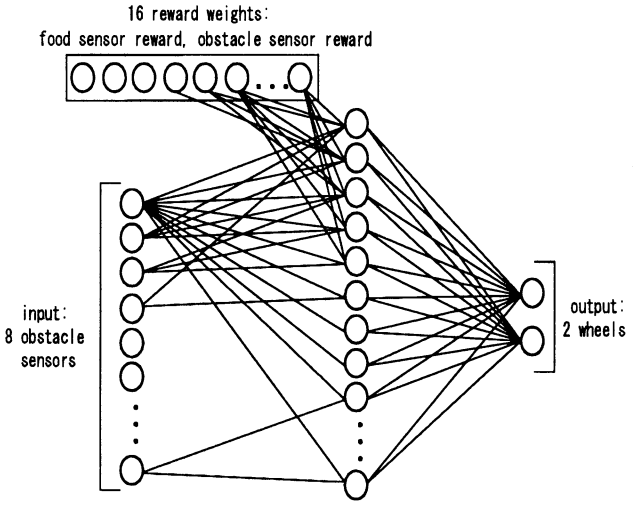


Fig.3 The neural network controller

3 Learning law for loitering

3.1 Structure of neural network controller

Figure 3 shows the neural network used for the learning of the loitering action. The back propagation neural network controller has eight inputs from the proximity sensors and has two outputs to the wheel motors, and has 16 hidden layer elements. In the learning, an action rule, which suits the sizes of environment and the robot, is learned.

3.2 Neural Network with Reward Weight Function

The reward is determined on the distance between the robot and the wall and the distance between the robot and the light so as to be correspondent to the bait and unexpected obstruction. Reward for the avoidance of obstruction is also given supplemental in order to respectively raise the learning for the obstruction avoidance for the 8 sensors. As shown in Figure 2, the infra red sensor returns smaller value as the robot approaches to the light source. As it is inconvenient for the reward value, we use the reciprocal of it to make reward values and adjust the sensor values so that its maximum value becomes 1 as follows:

$$\begin{cases} value\ of\ food & = & \dfrac{50}{light\ sensor\ value} \\ value\ of\ proximity & = & \dfrac{proximity\ sensor\ value}{1023} \end{cases}$$

The modefied light sensor value is shown in Figure 4.

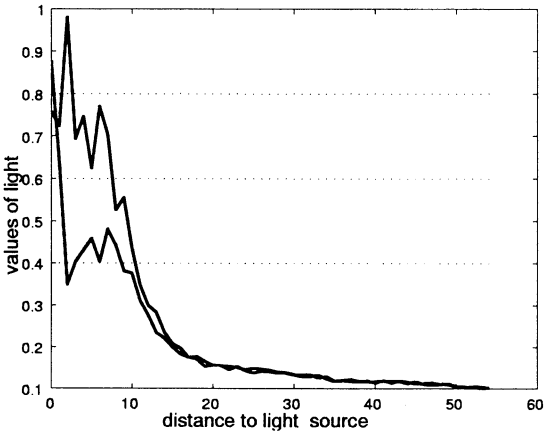


Fig.4 Modefied light sensor value

The evaluation of 5 stages from 0 to 4 in proportion to

the degree of the access to a bait is used as the reward value for acquisition of a bait and the reward value 4 is given when the robot gets the bait, which is shown in Figure 5. As to the reward for avoidance of obstacle,

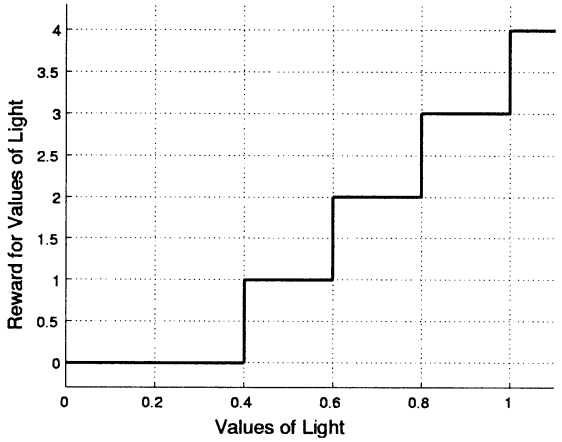


Fig.5 Reward value

-1 is given when the robot contacts with the obstacle. Such definition of the reward holds the divergence of the parameter as much as possible. The reward value is given to the hidden layer as following.

$$h_i(t+1) = C_i\ o_i(t)\ h_i^2(t) \qquad i = 1...8$$

$$h_{ij}(t+1) = \begin{cases} C_j\ r_j^2(t)\ h_i^2(t) & i = 9...14, j = 1...6 \\ C_j\ r_j(t)\ h_i^2(t) & i = 15, 16, j = 7, 8 \end{cases}$$

where $h_i(t)$, $h_{ij}(t)$ are weights of the hidden layer, $o_i(t)$ is the reward value for avoidance of the obstacle, $r_j(t)$ is and the reward value for acquisition of a bait and C_i, C_j are adjustable parameters. The reward unit in the hidden layer defines so that the robot may turn to the left, if a bait is in the left. When this reward unit is used by back-propagation neural network, the derivatives by r(t),o(t) and h(t) must be given for the calculation of the output error. When the reward values are given, these are inserted between input layer and hidden layer. When the robot notices the existence of the obstruction or the bait, it acts based on each reward. The outline of the system that summarizes the above is shown in Figure 6.

Combination of two learning rules defines the bait search action in which the loitering action is learned as a first step, and the acquisition action based on reward rule is learned in the after as a second step.

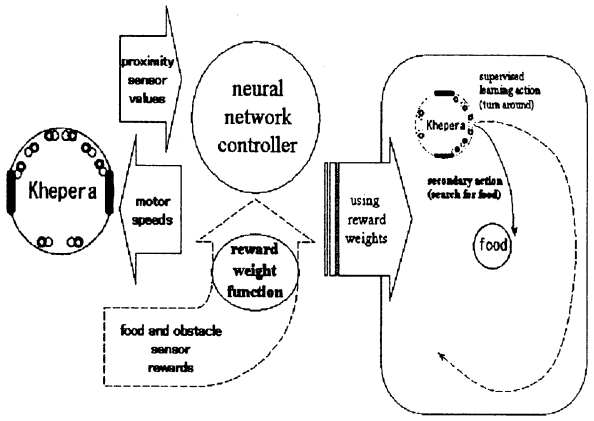


Fig.6 Proposed hybrid learning schema

4 Simulation and experiment

The simulation is carried out using the KiKS simulator with above mentioned environment where the Khepera robots and the baits represented by light sources are placed. The simulation is carried out by changing learning level and time.

Figure 7 shows the case of 500 teaching data, 300 learning epoch and 600 second experimental time.

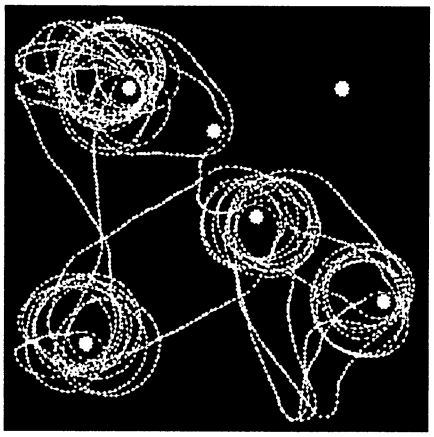


Fig.7 Trajectory of robot: learning epoch is 300, experimental time is 600 seconds

Figure 8 shows the case in which the back propagation is also used for the reward rule unit to raise the learning effect.

Comparison of the two cases shows that the search is widely carried out without remaining at a local field for former case, while in the latter case the search is often remains at a very narrow neighboring area of a bait and it occasionally leaves there, it stays again at a very narrow neighboring area of a different bait.

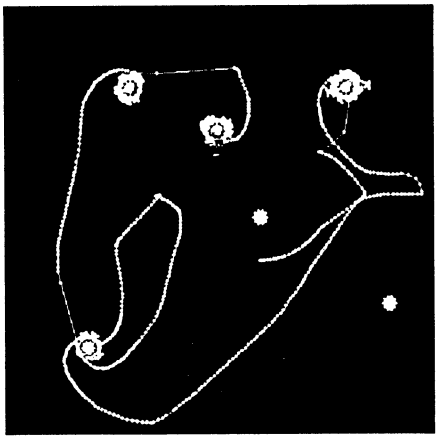


Fig.8 Trajectory of robot: learning epoch is 300, experimental time is 360 seconds, training goal is reached

As to the performance, large difference is produced in comparison with the case using the back-propagation and the case not using it. Figure 9 shows the difference in the attainment of the learning. In this experiment, the goal of the learning has be made to be 10^{-14} error, and the difference of the performance like the figure produces also large difference in the action. In the case of low attainment of the learning, the robot does not react to the bait and passes through the bait without stopping.

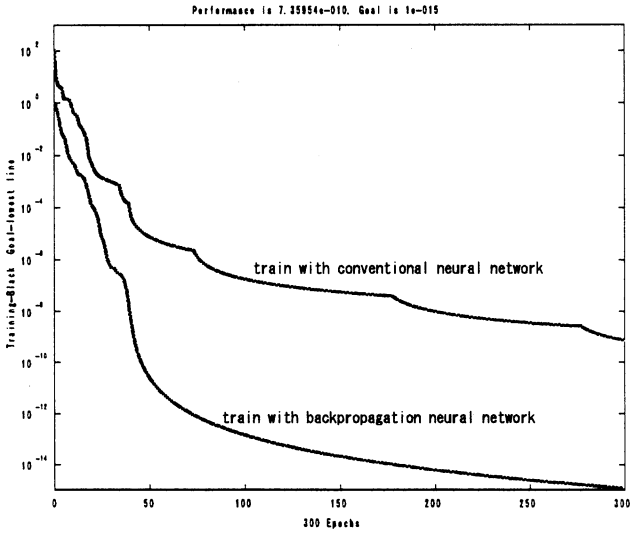


Fig.9 learning paformance

5 Summary

In this paper, bait acquisition method by neural network of a mobile robot is described. Combination of a supervised learning for the rough searching whereabouts of the bait and an unsupervised learning for acquiring a bait enables the robot to loiter around prey field and acquire many baits. It is found that in comparison with the case using the back-propagation and the case not using it, large difference is produced in performance of the robot behavior and in convergence of the learning. The increase of the learning frequency yields superfluous adaptation to the supervised learning and the robot cannot acquire the bait. It can be said that a balance of two learning rules is maintained by decreasing the teacher data. It seems to be better to take a large reward value, but it is apt to induce the explosion of the parameter of the neural network.

References

[1] I. Nishikawa and S. Uchida, "A Stochastic Unsupervised Learning Algorithm and Its Application to an Autonomous Agent," *Journal of the SICE*, Vol. 37, No. 12, pp. 1169-1177, 2001.

[2] H. Kimura, K. Miyazaki, S. Kobayashi, "A Guideline for Designing Reinforcement Learning Systems (in Japanese)," *Journal of the SICE*,Vol.38, No.10, pp.618-623, 1999.

[3] N. Ito, T. Kondo, K. Ito, "An Incremental Behavior Learning Based on Reinforcement Learning with Schema Extraction Mechanism for Autonomous Mobile Robot," *Proc. of SICE Annual Conf.*,TA II-12-1, 2003.

[4] K. Shibata, Y. Okabe, K. Ito, "Direct-Vision-Based Reinforcement Learning Using a Layered Neural Network," *Journal of the SICE* ,Vol. 37, No. 2, pp. 1-10, 2001.

[5] M. Iida, M. Sugisaka and K. Shibata, "Direct-Vision-Based Reinforcement Learning to a Real Mobile Robot," *Proc. of Int'l Conf. of Neural Information Proceeding Systems*,Vol. 5, pp. 2556-2560, 2002.

[6] K-Team SA, *Khepera USER MANUAL*, 1995.

[7] Theodor Nilsson, *KiKS is a Khepera Simulator*, http://www.kiks.net/.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A Genetic Algorithm for Multiple Face Detection

Xinjian Fan[1], Muhammad Rizon Bin Juhari[2], Masanori Sugisaka[1,3], and Hidenori Kimura[3,4]
[1]Department of Electrical and Electronic Engineering, Oita University
700 Dannoharu, Oaza, Oita 870-1192 Japan
(Tel: +81-97-554-7831; Fax: +81-97-554-7841) Email: {msugi, fxinjian}@cc.oita-u.ac.jp
[2]Computer and Communication School of Engineering, Northern Malaysia University College of Engineering, Malaysia
Email: rizon@kukum.edu.my
[3]The Institute of Physical and Chemical Research (RIKEN), Bio-Mimetric Control Research Center, Shimoshidami, Moriyama-ku Nagoya, 463-0003, Japan
[4]Complex Systems Department of Complexity Science and Engineering Graduate School of Frontier, The University of Tokyo, Tokyo 113-8656, Japan
Email: kimura@crux.t.u-tokyo.ac.jp

Abstract: In this paper, we give a genetic search method for multiple-face detection. Specifically, we use genetic algorithms (GAs) to search for subwindows that might contain a face in an image while applying a neural network based face filter to evaluate these subwindows. It involves iterating a traditional GA and maintaining the found solutions (faces) online. The individuals on subsequent iterations are forced to suffer from a penalty cost according to how much it is similar with these solutions. The similarity cost discourages the individuals to re-explore the area where a face has been found, thereby encouraging them to locate a face elsewhere. Test on a number of images have demonstrated the efficiency of the method.

Keywords: Face detection, neural network, genetic algorithms

1 Introduction

Face detection is a very important task since it constitutes the first step of a large area of applications: automated face recognition, advanced HCI (Human Computer Interaction), secure access control and person identification, etc [1]. Its accuracy and efficiency have a direct impact on the usability of the whole system. Given a single image or a sequence of images, the goal of face detection is to identify all image regions, which contain a face regardless of its three-dimensional position, orientation and lighting conditions.

Up to now there have been many face detection methods. The methods can be classified into two broad categories:

- Methods based on facial features or templates [2],[3].
- Methods based on face representations learned from a large number of examples (face images) using neural networks [4],[5] or statistical approaches: Support Vector Machines [6],[7], maximum likelihood [8], etc.

Generally, methods in the second category are more practical since they are robust against partial occlusion, excessive deformation and illumination changes. However, high computational cost introduced by exhaustive search at all possible locations and scales has prevented them to be a practical tool. In [9], we propose to use evolutionary algorithms to search the image efficiently based on a neural network face filter. Experimental results showed that the method achieved a great speedup compared to using the exhaustive search method.

The work described in [9] only focused on single-face detection. In this paper, we extend the method to detect multiple faces. Different from single-face detection, where we only need to find the maximal filter response to see if it is a face, the multiple-face detection problem is in fact a multimodal optimization problem (each modal corresponds to a face in the image). To locate these faces, we need to find all local maxima of the filter responses which are above a given threshold. However, traditional genetic algorithms (TGAs) are mainly known as robust tools for finding single, global optima. Although some techniques, such as fitness-sharing, provide one way to find multiple solutions, the time complexity of calculations is their disadvantages [10].

The method proposed in this paper is still based on a TGA, but with iterating it. In the iteration, the solution (face) of each run will be maintained online and the individuals on subsequent iterations are forced to suffer from a penalty cost according to how much it is similar with the solutions which have been found. The similarity cost discourages the individuals to re-explore the area where a face has been found, thereby encouraging them to locate a face elsewhere. The process can continue until we decide (using some criterion, such as the number of faces we expect) that all faces have been located. Test on a number of images have demonstrated the efficiency of the method.

The paper is organized as follows: In Section 2, we briefly introduce the work described in [9] - a neural network based face filter and the basic procedures of face detection using genetic search. Section 3 describes

in detail our extension of the genetic search method to the multiple-face detection problem. Finally, we present experiments and give the conclusion in Section 4.

2 Face detection using genetic search

2.1 Neural network based face filter

The face filter is used to evaluate a subwindow extracted from an image, to see how well it resembles a face. It should grasp as much as possible the common features of human face, while at the same time is not vulnerable to the background. In this paper, we use a retinally connected neural network (RCNN) [4] to serve as a face filter.

The neural filter takes as input a 400-length vector, which is transformed from a 20×20 subwindow. The input window is divided into 3 kinds of receptive fields: four 10×10 pixel subregions, sixteen 5×5 pixel subregions, and six overlapping 20×5 pixel subregions. Each of these 26 subregions has a full connection to two hidden neurons, for a total of 52 neurons in the hidden layer. There is only one neuron in the output layer, which generates a real value from -1 to 1, indicating to what extent a subwindow looks like a face.

The neural network is trained using back-propagation algorithm with a momentum term [11]. The momentum term can help to make the training faster. The weights of the neural network are updated only after a small set of examples, called a *mini-batch*, is processed. Each mini-batch has 50% positive examples and 50% negative examples, which are drawn randomly from the entire training sets.

The face training set is composed of 6000 frontal-view or nearly frontal-view faces (positive examples). 9548 random patches chosen from images containing no faces serve as initial non-face training set (negative examples). Additional non-faces were introduced by applying the bootstrap algorithm [4]. Both the face and non-face examples are enhanced by the preprocessing procedures such as lighting correction and histogram equalization.

2.2 Genetic search

The flowchart of the genetic search method is shown in Figure 1. In our problem, each individual represents a subwindow within the input image, which is encoded by its size and position. The initial population is randomly generated. To evaluate subwindows of different sizes, we rescale each subwindow into a standard size. After the preprocessing procedure, the standardized subwindow is sent to the neural network based face filter. The larger the value of its output, the more the sub-window resembles a face. So the subwindow with a high output value will be given a high fitness. Three genetic operators, selection, crossover and mutation, guide the search to possible face regions in the image. A face is indicated when the best individual's filter response is above a predefined threshold.

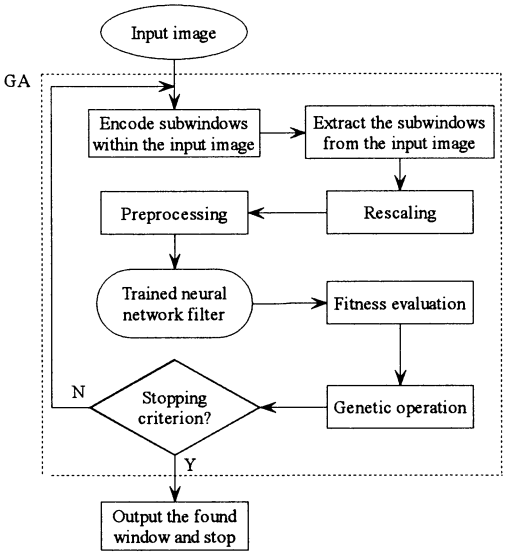


Fig. 1 System diagram

2.2.1 Encoding

In our problem, each individual in the population represents a subwindow in the image. In this paper, we use the center (C_x, C_y) and length S to define a subwindow (Figure 3). To evaluate subwindows of different sizes using the neural network, we should rescale them to the size of 20×20 (the input size of the neural network). However, if this computation is done on every size of subwindows, it will be very time-consuming. To avoid it, we first build an image pyramid as in the exhaustive search method:

$$\left\{INT\left(\frac{W}{q^k}\right)\times INT\left(\frac{H}{q^k}\right)\right\} \quad k:\ 0\rightarrow L \tag{1}$$

where W and H are the width and height of the input image respectively, and q is the scale factor, in our experiment, $q=1.2$. As the programming language C, INT means taking the integer part of a value. The top level (level L) should have a size more than 20×20:

$$\frac{MIN(W,H)}{q^L}\geq 20\text{, gives } L=INT\left(q\sqrt{\frac{MIN(W,H)}{20}}\right)$$

Then we let S to be chosen among the following geometric sequence:

$$\{INT(20q^k)\} \tag{2}$$

For a subwindow $\mathbf{F}_w=(C_x,C_y,20q^k)^T$, we find its mapped 20×20 window $\mathbf{F}'_w=(C'_x,C'_y,20)^T$ in level k of the pyramid (See Figure 2) by:

$$C'_x=INT(\frac{C_x}{q^k}),\ C'_y=INT(\frac{C_y}{q^k}) \tag{3}$$

So the chromosome of an individual consists of 3 genes, represented by the data structure shown in Figure 3. C_x, C_y and k are defined in [10, W-10], [10, H-10] and [0, L] respectively.

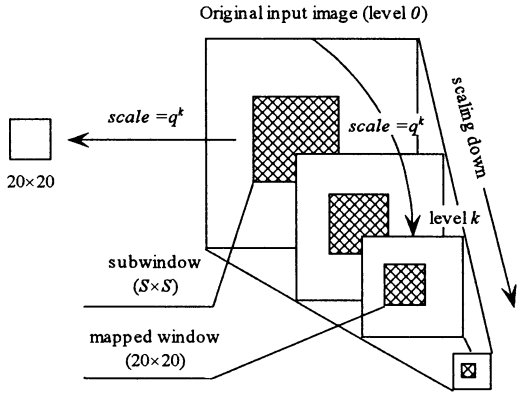


Fig. 2 For each subwindow, its corresponding 20×20 input window is found in the image pyramid.

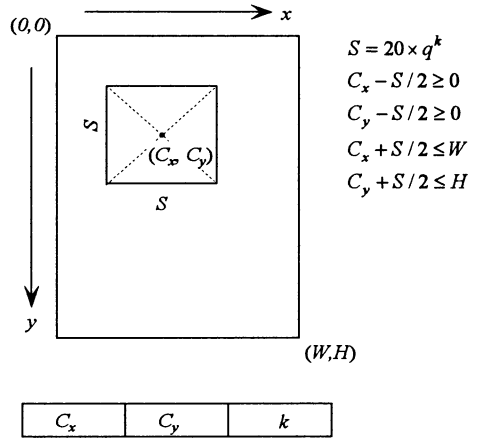


Fig. 3 Chromosome representation and constraints

2.2.2 Fitness evaluation

The optimization target in our problem is to maximize the detection value (the output of the neural network corresponding to a subwindow): the larger its detection value (*dv*), the more a subwindow resembles a face. The fitness function $f(\mathbf{F}_w)$ is given as

$$f(\mathbf{F}_w) = 0.5 \times (1 + dv(\mathbf{F}_w)) \quad \mathbf{F}_w \in \mathbf{T} \tag{4}$$

where $\mathbf{F}_w$ is a subwindow, $dv(\mathbf{F}_w) \in [-1, 1]$.

2.2.3 Genetic design

We use a real-coded GA (RGA), which operators and parameters are set as summarized in Table 1.

3 Multiple face detection

In this section we extend our GA search method to multiple face detection. The algorithm works by iterating the above traditional GA, but uses knowledge gained during one iteration to avoid re-searching regions of search space where solutions have already been found. This is achieved by maintaining the solution found in each run online and having individuals on subsequent iterations suffer from a similarity cost with these solutions. So fitness values are depressed near the area where a face has been found, thereby making individuals to locate a face elsewhere.

Table 1. Settings for RGA

Crossover operator	BLX-α crossover [12]
Mutation operator	Gaussian mutation [12]
Selection	Linear rank selection with elitism
Size of population	100
Maximum generation	100
Probability of crossover	0.90
Probability of mutation	0.15

For this purpose we need a distance metric. This is simply a function which, given two individuals, returns a value related to how "similar" they are to each other. Here we use the normalized separation of parameters, given by equation 5,

$$D(x,y) = \sum_{i=1}^{N} K_i \times \frac{|x_i - y_i|}{b_i - a_i} \tag{5}$$

where N is the number of individual parameters, x_i and y_i are the i^{th} parameter in the x and y individual respectively, k_i is the similarity weight of the i^{th} parameter, and b_i and a_i are the upper and lower bounds of the i^{th} parameter space.

The distance metric between an individual x and an optimal solution which has been found s must be greater than R_c, the critical radius of separation, for the individual in the next run to continue to next generation. Thus the similarity cost function is defined as

$$G(x,s) = \begin{cases} 1 & D(x,s) < R_c \\ 0 & otherwise \end{cases} \tag{6}$$

Having defined a distance metric and similarity cost function, the algorithm works as follows (terms in italics are explained below):

Step 1: Initialize the population and equate the *modified fitness function* with the *raw fitness function*;

Step 2: Run the GA, using the modified fitness function;

Step 3: If the detection value of the best individual exceeds the threshold, display this as a face and keep a record of it. Otherwise, go to step 5;

Step 4: Update the modified fitness function to give a depression in the region near the best individual, producing a new modified fitness function;

Step 5: If the *stop criterion* is met, exit. Otherwise,

re-initialize the population and return to step (2).

Here the **raw fitness function** is referred to the function given in equation 4, $f(x)$;

The **modified fitness function**, $f_n(x)$, for an individual x, is computed from the raw fitness function and a number of similarity cost functions. Initially we set $f_0(x) \equiv f(x)$. At the end of each run, the best individual, s_n, found in that run is used to determine a similarity cost function, $G(x, s_n)$. The modified fitness function is then updated according to:

$$f_{n+1}(x) \equiv f_n(x) * (1 - G(x, s_n)) \quad (7)$$

The **stop criterion** can be set as the number of faces we expect has been achieved or some fun fails to find a new face.

4 Experiments and conclusion

We have tested the above algorithm on 20 images with different illumination conditions and increasing complexity background. All of the images have a size of 320×240 and there are two or three faces in each image. There are 48 faces totally in the test set. Here we show one of them (Figure 4). During experiment, we assume that we know, or can estimate the number of faces in the image. The stop criterion is set as: the GA has found the expected number of faces or fails in two runs. Among the 48 faces, the GA succeeded to find 45 faces (a success rate of 93.8%). The average processing time is about 1s (the shortest is 0.35s for 2 faces and the longest is 1.98s for 3 faces) on a Pentium 900MHz PC. Considering the exhaustive search will take about 22s, the proposed method achieved a great speedup.

Clearly the more faces there are in the input image, the more time consuming for the GA to find all the faces. So the method is fit for face detection without too many faces in the input image.

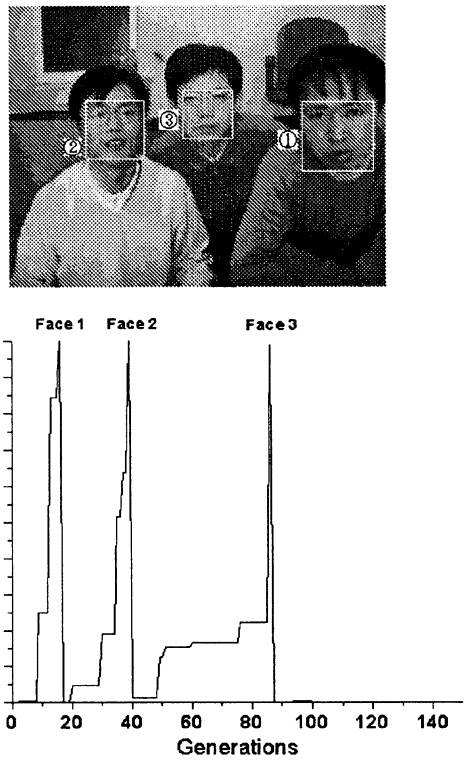


Fig. 4 A scene and GA's performance plot

References

[1] Ming-Hsuan Yang, David Kriegman and Narendra Ahuja, "Detecting faces in images: A survey", IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), vol.24, no.1, pp.34-58, 2002.

[2] Huang C. and Chen C. Human, "Facial feature extraction for face interpretation and recognition", Pattern Recognition, 25(12): 1435-1444, 1992.

[3] Brunelli R. and Poggio T., "Face recognition: Features versus templates", IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(10): 1042-1052, 1993.

[4] Henry A. Rowley, "Neural Network-Based Face Detection", Thesis submitted for the degree of Doctor of Philosophy, School of Computer Science, Carnegie Mellon University, 1999.

[5] B. Fasel, "Fast multi-scale face detection", IDIAP-COM 98-04, IDIAP, 1998.

[6] E. Osuna, R. Freund, and F. Girosi, "Training support vector machines: An application to face detection", Computer Vision and Pattern Recognition, 1997.

[7] S. Romdhani, P. Torr, B. Scholkopf, A. Blake, "Computationally efficient face detection", Proc. of ICCV 2001, pp.695-700, Vancouver, Canada, 2001.

[8] H. Schneiderman and T. Kanade, "A statistical method for 3D object detection applied to faces and cars", Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.746-751, 2000.

[9] M. Sugisaka, X. Fan, "Fast Face Detection Using Genetic Algorithms and Pyramid Structure", Proc. of the 8th Int. Symp. on Artificial Life and Robotics, vol.2, pp. 160-163, Oita, Japan, 2003.

[10] D. Beasley, DR Bull, and RR Martin, "A sequential niche technique for multimodal function optimization", Evolutionary Computation, vol.1, pp.101-125, 1993.

[11] Hertz, J., Krogh, A. and Palmer, R. G, "Introduction to the Theory of Neural Computation", Addison-Wesley Publishing Company, Redwood City, 1991.

[12] F. Herrera, M. Lonzano, and J. L. Verdegay, "Tackling real-coded genetic algorithms: Operators and tools for behavioral analysis", Artificial Intelligence Review, 12(4), 1998.

[13] Goldberg D., "Genetic algorithms in search, optimization and machine learning", Addison-Wesley, pages Reading, MA, 1989.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Global Identification of Complex Systems

J. Swiatek
Institute of Control and Systems Engineering,
Wroclaw University of Technology,
wyb. Wyspianskiego 27,
50-370 Wroclaw, Poland
e-mail: jerzy.swiatek@pwr.wroc.pl

Keywords: Identification, modeling, complex systems

Abstract

In the paper some problem of input output complex systems identification is presented. The description of complex system is given by description of each system element and structure. The local and global identification problems are formulated. The different approach to the identification problem is discussed. Base on the multi-criteria concept the globally optimal model with respect to quality of local models is presented.

1 Introduction

Design of computer aided management systems for complex plant generates new tasks of complex system identification. The similar problems can be found designing computer control systems for complex process or modeling of complex systems different nature (for example: biological plants). Usually in such a system the elementary processes are defined and connection between elements is given. In the production process the elementary operations are elements of complex systems and structure of the system is given by time schedule of operations given by technological conditions. In this case we have problem of modeling of complex of operation system [1,3]. Investigation of technical or biological plants [1,4,5,6] usually gives us modeling identification of complex input-output system. In this case we can distinguish sub process (elementary process) with some inputs and outputs which can operate separately. Connection between inputs and outputs of each element give us complex system. Modeling of such a system is connected with modeling of each process separately taking into account connection between elements. Such a problem formulation gives us new tasks of identification of complex systems. Some times in such a system some inner inputs and outputs could no be measured. We can have problem of identification with restricted measurement possibilities [6]. Another problem we have analyzing two level (generally multi-level) management or control process. Dependently on the level we expected model of the different quality. On the lower lever it is better to prepare locally optimal model, but on upper level globally optimal description required. In this paper the local and global identification problem is discussed. The multi-criteria approach is used to formulate the globally optimal model.

2 Description of the complex system

Let us consider input output complex system consists on M subsystems $O_1, O_2, \ldots, O_M$. Let:

$$\bar{y}_m = \Phi_m(u_m, a_m) \tag{1}$$

denotes m-th system elements model with input u_m and output y_m, where: $\bar{y}_m$ - vector of model output, a_m - vector of model parameters of m-th element, $m=1, 2, \ldots, M$. Input, output and model parameters are vectors from respective spaces, i.e.:

$$u_m \in U \subseteq \Re^{S_m}, \quad y_m, \bar{y}_m \in Y \subseteq \Re^{L_m}, \quad a_m \in A \subseteq \Re^{R_m}.$$

Let u, y, $\bar{y}$, a denotes vector of all inputs, pant and model outputs and model parameters, respectively, i.e.:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_M \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix}, \quad \bar{y} = \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_M \end{bmatrix}, \quad a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix}.$$

The structure of the complex system is given by connection between inputs and outputs of each element. It is necessary to define which input of particular element is

output of the other element. Let us note that some system input are not output of any other elements, see Figure 1. Those inputs are called external inputs – x. Further more let us distinguish some plant

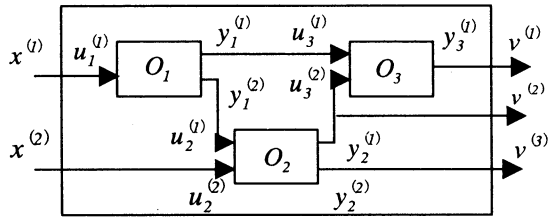


Figure 1. Example of complex system

outputs, which are important for model application, for example outputs which are used to formulate control problem on the upper level. Those outputs we call global outputs – v. Now structure of the system may be described by the following equations:

$$u = Ay + Bx, \tag{2}$$

$$v = Cy, \tag{3}$$

where: A, B are zero-one coincidence matrix and matrix C shows global outputs. On the figure 1, the example of the complex system is given. The system consists on elements O_1, O_2 and O_3.The external inputs x, global output v and respective inner connections are illustrated. The equation (2) and (3) for system in figure 1, have forms:

$$\begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \\ u_2^{(2)} \\ u_3^{(1)} \\ u_3^{(2)} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_1^{(1)} \\ y_1^{(2)} \\ y_2^{(1)} \\ y_2^{(2)} \\ y_3^{(1)} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x^{(1)} \\ x^{(2)} \end{bmatrix}, \tag{4}$$

$$\begin{bmatrix} v^{(1)} \\ v^{(2)} \\ v^{(3)} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} y_1^{(1)} \\ y_1^{(2)} \\ y_2^{(1)} \\ y_2^{(2)} \\ y_3^{(1)} \end{bmatrix}. \tag{5}$$

Let:

$$\bar{y} = \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_M \end{bmatrix} = \begin{bmatrix} \Phi_1(u_1, a_1) \\ \Phi_2(u_2, a_2) \\ \vdots \\ \Phi_M(u_M, a_M) \end{bmatrix} = \Phi(u, a). \tag{6}$$

Substituting (2) in the place of u in (6) we have:

$$\bar{y} = \Phi(A\bar{y} + Bx, a). \tag{7}$$

Solution (7) with respect to $\bar{y}$ is:

$$\bar{y} = \Phi^{-1}(x, a; A, B). \tag{8}$$

Taking into account (3) we have:

$$\bar{v} = C\Phi^{-1}(x, a; A, B) = \bar{\Phi}(x, a). \tag{9}$$

The equation (9) is the global model of complex with external inputs x and global outputs v.

3 Complex system identification

To determine the optimal model parameters of complex system the measurements of input and output each element are collected. Choice of the best model parameter in this case is reduced to determination such a values of parameters for which identification performance index is minimal. The performance index is a measure of difference between model outputs and respective measurements. Because of the complexity of the system there are different possibilities to define identification performance index. On the one hand it is possible to determine optimal model for each system element separately and on the other hand we can define performance index for the system as a whole. In the first case we obtain the locally optimal models for each element separately not taking into account structure of system. The second case gives the globally optimal model for the whole system.

3.1 Locally optimal model

Taking into account experiment for each element separately the following local identification performance index may be defined:

$$Q_m(a_m) = \sum_{n=1}^{N} q_m\left(y_{m,n}, \Phi_m(u_{m,n}, a_m)\right), \quad m = 1, 2, \cdots, M, \tag{10}$$

where: q_m is measure of difference between output of the m-the plant and its model output (see Figure 2.), $u_{m,n}$, $y_{m,n}$ are n-th measurements of m-th element input and output, N is number of measurements.

Locally optimal parameters can be obtained by minimization identification performance index (10) for m – th element with respect a_m, i.e.:

$$a_m^* \to Q_m(a_m^*) = \min_{a_m} Q_m(a_m) \quad m = 1, 2, \cdots, M. \tag{11}$$

Equations (1) and (6) with parameters a_m^* obtained in (11) are locally optimal models of the separate elements. After composition equation (9) with parameters a_m^* is a locally optimal model of the system as a whole.

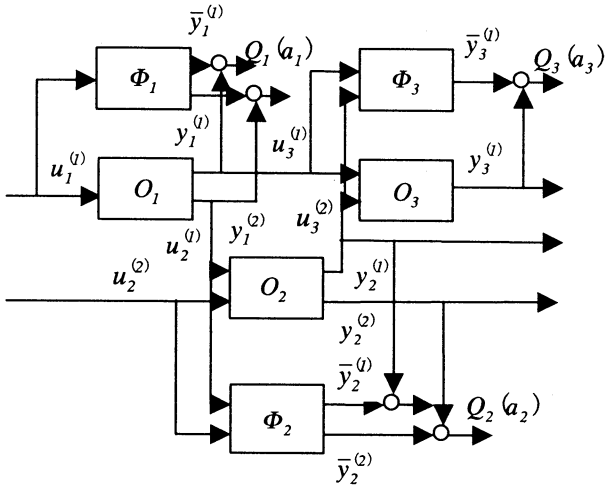


Figure 2. Complex system with locally optimal models

3.2 Globally optimal model

Another view on the system, i.e. comparison only selected plant output gives with global model output defines the globally optimal model, see Figure 2. In this case global identification performance index measure the difference between selected plant output and global model output v. The global identification performance index has the form:

$$Q(a) = \sum_{n=1}^{N} q\left(v_n, \overline{\Phi}(x_n, a)\right), \tag{12}$$

where: q is a measure of difference between selected output of complex plant and respective model global outputs (see Figure 3.), v_n, x_n denote the n-th measurements of global plant outputs and external inputs, respectively.

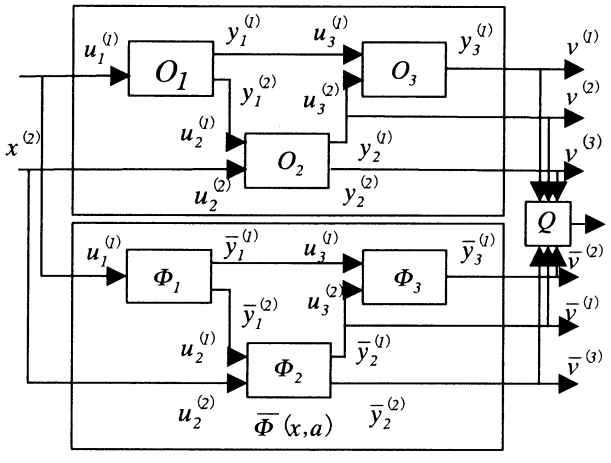


Figure 3. Complex system with globally optimal model

The globally optimal parameters of complex system model - a , are obtained by minimization of performance index (12) with respect a, i.e.:

$$a \to Q(a) = \min_{a} Q(a). \tag{13}$$

Equations (1) and (6) with parameters a obtained in (13) are globally optimal models of the separate elements. Equation (9) with parameters a is a globally optimal model of the complex system as a whole.

3.3 Globally optimal model with respect to quality of local models

The presented approach shows two different locally and globally optimal models. There is a question: Which one is better? For example, if proposed model will be used to investigate control algorithms for complex system then the answer depends on the structure of control systems. In two levels control system investigation of control algorithms on the lower level requires locally optimal models, but on the upper level the globally optimal model is necessary. There are two different criteria: local (10) and global one (12). There is the possibility to formulate the other approaches based on multi-criteria approach. One of them is to define new synthetic performance index which take into account both local and global model quality. It can have the form:

$$\overline{Q}(a) = \alpha_0 Q(a) + \sum_{m=1}^{M} \alpha_m Q_m(a_m) \tag{14}$$

where: $\alpha_0, \alpha_1, \cdots, \alpha_M$ sequence of weight coefficients. They shows weigh of participation of global (12) and local (10) indexes respectively, in the synthetic performance index. So now the optimal model parameters for synthetic performance index

$$\overline{a} \to \overline{Q}(\overline{a}) = \min_{a} \overline{Q}(a) \tag{15}$$

In the other approach we assume that local models must be sufficiently good. It means that the local performance indexes must be less the given number. So parameters must fulfill the following conditions:

$$Q_m(a_m) \le \beta_m, \quad m = 1, 2, \cdots, M \tag{16}$$

where quality sufficient number β_m is grater then locally optimal performance index, i.e.:

$$\beta_m > Q_m(a_m^*), \quad m = 1, 2, \cdots, M. \tag{17}$$

Now, the optimal model parameters will be obtained by minimization performance index (12) with constrains (16).

4 Cascade structure

In practical applications very popular case of complex system is a system with cascade structure (Figure 4.). To the mentioned structure can be transferred all complex systems without feedbacks.

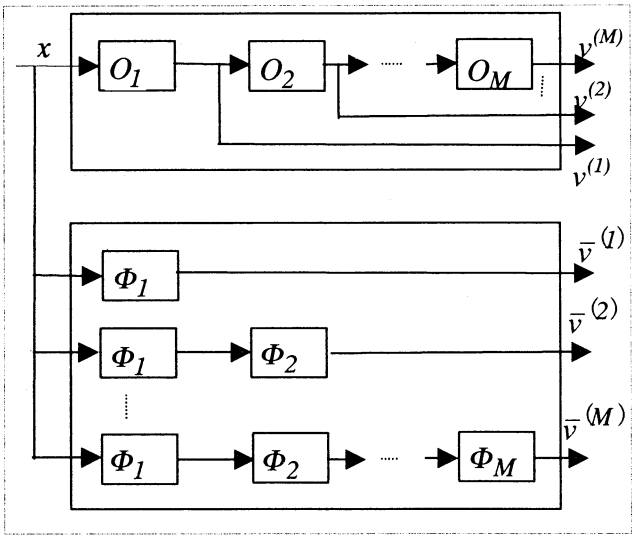


Figure 4. Complex system and global model with cascade structure

The respective global model (9) for this structure has the form:

$$\begin{bmatrix} \bar{v}^{(1)} \\ \bar{v}^{(2)} \\ \vdots \\ \bar{v}^{(M)} \end{bmatrix} = \begin{bmatrix} \Phi_1(x,a_1) \\ \Phi_2(\Phi_1(x,a_1),a_2) \\ \vdots \\ \Phi_M(\cdots\Phi_2(\Phi_1(x,a_1),a_2)\cdots a_M) \end{bmatrix} \quad (18)$$

The global identification performance index (12) is:

$$Q(a) = \sum_{n=1}^{N}\sum_{m=1}^{M} q(v_n^{(m)}, \bar{v}_n^{(m)}) \quad (19)$$

Notice that the model (18) may be given in the recursive fom:

$$\bar{v}^{(m+1)} = \Phi_m(\bar{v}^{(m)}, a_m), \quad m = 0,1,\cdots,M \quad (20)$$

where:

$$\bar{v}^{(0)} = x.$$

To determine the globally optimal model parameters the dynamic programming methods may be used.

5 Final remarks

In the paper the problem of complex input-output systems is discussed. The locally and globally optimal modes have been introduced. It has been shown that based on multi-criteria approach, the other models may be defined. It has been shown that for cascade structure the dynamic programming approach may be used. The presented approach is useful for investigation of computer control systems for complex plant.

References

[1] Bubnicki Z., *Identification of Control Plants*, Oxford, N. York, Elsevier, 1980.

[2] Bubnicki Z., *Problems of complex systems identification,* Proc. of International Conference on Systems Engineering, Lanchester Politechnic, Coventry, 1980.

[3] Bubnicki Z., *Global modeling and identification of network systems,* Proc. of 3rd International Conference on Systems Engineering, Wright State University, Dayton, USA, 1984.

[4] Dra³us G., Swiatek J., *Global modeling of complex systems by neural networks,* Proc. of 7th International Symposium on Artificial Life and Robotics, Oita, Japan, 2002.

[5] Swiatek J., *Two stage identification and its technical and biomedical applications,* Wydawnictwo Politechniki Wroc³awskiej, Wroc³aw, 1987 (in polish)

[6] Swiatek J., *Identification,* Problems of Computer Science and Robotics. Grzech A., editor, Zak³ad Narodowy im Ossoliñskich – Wydawnictwo PAN, Wroc³aw, 1998. (in polish)

[7] Swiatek J. *Global and local modeling of complex input output systems.* Proc. of Sixteen International Conference on Systems Engineering, September 9-11 2003, Coventry University, Coventry, England pp. 669-671.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

An Experiment of Dissimilarity Reconstruction in Information Recommendation

Kou Zhongbao , Ban Tao, Zhang Changshui*, Zhuang Like
the State Key Laboratory of Intellegent Technology and Systems
Department of Automation, Tsinghua University, Beijing 100084, P.R.China
E-mail: * zcs@mail.tsinghua.edu.cn

Abstract— **In this paper, we use a technique named Dissimilarity Reconstruction (DSR) on a text categorizting set. According to DSR, objects in information recommendation are represented in an intrinsic dimensionality space of the data. In the space, distance between associated vectors of two arbitrary objects well represents the dissimilarity between them in sense of evaluation. We also make some discussion on the size the data set and generalization of DSR.**

Keywords— **information recommendation, dissimilarity reconstruction, Isomap, intrinsic dimension, geodesic distance**

I. Introduction

We have put forward a technique named Dissimilarity Reconstruction (DSR) to discover relationships between the interests using data from Bulletin Board System (BBS) in [1]. In this paper, we apply the technique on a text categorizting set to illustrate the utility of DSR in information recommendation [2].

DSR tries to grasp the intrinsic structure of the data set and maps the data into its intrinsic dimensional space. Then what is the intrinsic structure of a data in a information recommendation system?

Take navigations on the World-Wide-Web as an example: when we are browsing a web page, we often follow the hyperlinks in the page to a new page. After shuttling for dozens of times, the page we finally arrive at may have nothing to do with the one we started with. This gradually changing and transferring process in information evaluation may well reflect the underlying structure of the data. DSR just takes the approach of simulating the mechanism in evaluation and building relationships between objects according to the transferences.

The remainder of this paper is organized as follows. Section II demonstrate the experiment using DSR and discuss its results. In Section III, discussions on influence of the size of the data set and generalization of DSR are given. Concluding remarks are given in Section IV.

II. Experiment

A. Data

Data set adopted in the experiment is the "Reuters-21578" text categorization test collection. The documents in this collection appeared on the Reuters newswire in 1987. There are several reasons to choose such a standard and widely distributed data set:

1. Document formatting and manually text categorizting were already done on this well formatted and indexed data set. Then, experiments on such an open data set are easy to be tested.
2. The "topics" categories of this data set are economic subject categories. Examples include "coconut", "gold", "inventories", and "money-supply". These specific categories are suitable for us to explain relationships among topics.
3. For documents published on the newswire, the topic of a document always has direct relation with its glossary. This feature makes the TFIDF representation reasonable.

Several subsets from "Reuters-21578" collection are chosen to perform the experiment.

To show the distribution of documents in different topics, three topics "coffee", "Consumer Price Index (CPI)" and "nature gas" are chosen. This data set will be referred as subset 1 in later discussion. The numbers of documents that the three topics contain are 144, 112, 130 respectively, which add up to 384 documents. Two of the documents share the topics of "CPI" and "nature gas", while others do not have more than one topic.

To show the distribution of documents in a single topic, the 112 documents in the topic named "CPI" are chosen, which will be referred as subset 2 in the following discussion.

B. DSR to the Data

B.1 VSM representations of the data

Follow the steps below to represent the data in VSM.
1. Data preprocessing: extract the keywords and frequencies from each of the documents. Here, we identify different tenses and forms of the same word as a single keyword.
2. Obliterate the words with extremely rare occurrence. Such words usually serve as source of noise.
3. Wipe off the words that only appear in a particular document. Such words do not contribute to the relation of two objects and thus can be deemed as another kind of noise.
4. Build the VSM presentation of the data set, where weight of each item is computed based on TFIDF [3]. When only words appearing more than 10 times in each subset are kept, the documents in subset 1 and in subset 2 are represented as 738 and 201 dimensional vectors respectively.

B.2 Dissimilarities between objects

Then Compute the similarity of two objects $sim(k_1, k_2)$ by

$$sim(k_1, k_2) = \frac{\bar{v}_{k_1} \cdot \bar{v}_{k_2}}{\sqrt{|\bar{v}_{k_1}|_2 \cdot |\bar{v}_{k_2}|_2}} = \frac{\sum_{i=1}^{m} (v_{k_1,i} \cdot v_{k_2,i})}{\sqrt{(\sum_{i=1}^{m} v_{k_1,i}^2) \cdot (\sum_{i=1}^{m} v_{k_2,i}^2)}}, \quad (1)$$

where k_1,k_2 denote two arbitrary objects, $\bar{v}_{k_1}$,$\bar{v}_{k_2}$ are their associated vectors respectively, and m is the dimensionality of the vectors.

From $sim(k_1, k_2)$, we derive the dissimilarity between two objects $D(k_1, k_2)$ by

$$D(k_1, k_2) = 1 - sim(k_1, k_2). \quad (2)$$

Obviously, we get $D(k_1, k_2) \in [0, 1]$.

Compare with similarity, dissimilarity is more convenient in later computation and statement.

B.3 Dimensionality reduction with Isomap

Figure. 1 shows the dimensionality reduction of Isomap [4]. Fig. 1a is an intrinsic two-dimensional S-like surface in the three-dimensional space. Instead of using Euclidean distances in the three-dimensional space, Isomap computes the geodesic distances between pairs of samples in the data set. In this way, Isomap maintains the intrinsic structure of the data and scatters the samples into a two-dimensional space. Fig. 1b shows the two-dimensional embedding recovered by Isomap, which preserves the geodesic distances.

We have got the dissimilarity array $D_{n \times n}$, where $D(k_1, k_2)$ is calculated by Eq. (1) and Eq. (2). Then, we map the objects into a low-dimensional space following the main procedure of Isomap [4]:

1. Determine which objects are neighbors based on dissimilarities between pairs of objects, and build a graph, where vertices are the objects and edges are constructed by connecting each object to all of its k nearest neighbors;
2. Calculate the geodesic distances between all pairs of objects by summing up the dissimilarities between neighboring objects along the shortest path in the graph;
3. Construct an embedding of the data in low-dimensional space by classical MDS based on the geodesic distances.

The dimensionality of the reconstructed space are chosen as the intrinsic dimensionality of the data set, which is determined from the Residual Variance (RV)-dimensionality curve of Isomap or DSR. In this way, we get the mapping from high dimensional space to its intrinsic dimensional space.

C. Results

Fig. 2a shows the RV-dimensionality curves of DSR on the two subsets, where the "elbows" are evident in both cases. We can estimate the intrinsic dimensionalities of the two subsets as 6 and 4 respectively. Thus, DSR has found mappings from very high dimensional spaces to rather low ones.

To illustrate the distribution of the documents, Fig. 2b and Fig. 2c gives the two-dimensional projections derived by DSR for the two subsets. The RVs of the two cases are 0.279 and 0.286 respectively. Despite the errors caused by the projection, the figures show interesting features.

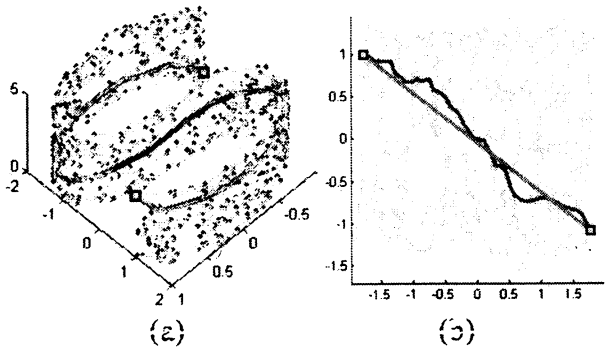


Fig. 1. Using Isomap to capture the underlying global geometry of a data set. (a) An intrinsic two-dimensional S-like surface in the three-dimensional space. Black segments show the geodesic path between two points (marked with squares). Gray lines between samples show the neighborhood graph built by connecting each point with its k (here k=7) nearest neighbors. (b) Two-dimensional embedding of the S-like surface in (a) by Isomap. Black segments show the geodesic path in two-dimensional space. Length of bold gray line is the Euclidean distance in the two-dimensional space.

In Fig. 2b, clusters are obviously formed with respect to the topics of the documents, which are consistent with the manually categorization of "Reuters-21578" collection. Such clusters indicate that Euclidean distance between two documents in low dimensional space reconstructed by DSR gives a good measure of their discrimination in semantic sense. Here, we take the documents 10567 and 17161 for further discussions, which are marked with circles in Fig. 2b and with titles and keywords listed in Table I. Both the two documents focus on energy consumption and price fluctuation of energy resources, so the very small distance between them is proper. Price of nature gas is also mentioned in these two documents, so it seems reasonable that they locate at the boundary of the two clusters and are more adjacent to the "CPI" cluster.

Fig. 2c illustrates the two-dimensional projection of the documents in subset 2. Although these documents have the same topic of "money-supply", clusters can be found in Fig. 2c. By assigning different marks to different types of documents, some interesting phenomena emerge: documents in "Cluster 1" appear as brief news without contents but titles, however the rest all have detailed contents. From Fig. 2b together with Fig. 2c, it seems that DSR not only is able to form clusters based on text contents but also may find some sub-clusters with respect to document formation or some other criteria.

To examine the transference in semantic sense, a path is marked out in Fig. 2c, which goes from document 10377 to document 17161, bypassing "10261" and "10567". Titles, keywords and frequencies of keywords of concerned documents are listed in Table I. Similarities between pairs of adjacent documents along the path can be: "10377" and "10261" share the same title, but the former is a brief one that makes some comparison among figures in succeed-

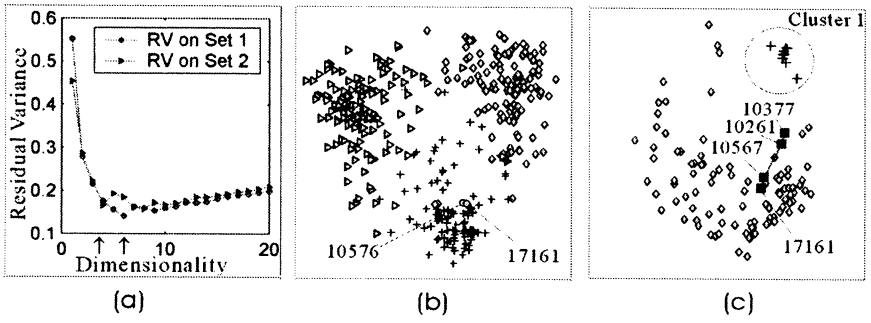


Fig. 2. DSR on "Reuters-21578" collection. (a) The RV-dimensionality curves of DSR on two subsets. Arrows mark the intrinsic dimensionalities acquired. (b) The two dimensional projection by DSR for subset 1 (in this case, the residual variance is 0.279) where diamonds, crosses and triangles stand for documents with topics of "coffee", "CPI" and "nature gas" respectively. The two documents marked with circles share the topics of "CPI" and "nature gas", while others do not have more than one topic. Labels of the two documents are their IDs in "Reuters-21578" collection. (c) The two-dimensional projection of subset 2 (in this case, the residual variance is 0.286), where diamonds denote typical documents which have both titles and bodies of contents and crosses denote documents which appear as brief news without bodies of contents. A transference path is also showed in (c), documents concerned in the path are labeled and marked with black squares.

TABLE I
INFO OF SOME FILES IN TOPIC CPI

ID	Document Title	Keywords and Frequencies
10377	JAPAN CONSUMER PRICES UNCHANGED IN FEBRUARY	Pct-5[a)], index-4, drop-4, early-3, February-3, Japan-2
10261	JAPAN CONSUMER PRICES UNCHANGED IN FEBRUARY	Price-8, pct-7, index-6, year-6, February-5, drop-5, early-4, consume-3, vegetable-3, fall-3
10567	U.S. ENERGY COSTS ROSE IN FEBRUARY BY 1.9 PCT	Pct-11, February-7, rise-7, energy-4, cost-4, index-4, department-3, price-3, say-3, gas-3
17161	U.S. ENERGY COSTS ROSE IN MARCH BY 1.0 PCT	Pct-11, rise-8, March-7, energy-4, cost-4, index-4 gas-3, department-3, price-3, say-3

a) The frequency of a keyword in a document is denoted by a number following the keyword.

ing months, while the latter details in some other aspects; "10261" and "10567" share the same writing style of detailing the price fluctuations of different goods in markets; and as mentioned in preceding discussion, both "10567" and "17161" discuss energy consumption and price fluctuation of energy resources. Thus, DSR discovers the internal relationships among documents in content-based information recommendation too.

III. Discussion

In this section, we will give some discussions on size of data set and potential applications of DSR.

A. Discussion on Size of Data Set

Since the steps of Isomap, including neighborhood graph constructing, geodesic distance calculating and MDS, have close relations to the data set, DSR is also influenced by the data set.

Fig. 3a and Fig. 3b are two planes acquired from the S surface in Fig. 1a. Line 3 in Fig. 3a is the ideal geodesic path between two points A and B on the S surface. Fig. 3b shows the cutaway view of the geodesic path, where curve 3 is identical to line 3 in Fig. 3a. When samples on the S surface are sparse, error will occur in geodesic distance calculating. There are two possible sources of errors: firstly, departure of samples from ideal geodesic path on the S surface, as Fig. 3a shows; secondly, difference between Euclidean distances and distances along the S surface between neighboring points, as Fig. 3b shows. But as the number of samples increases, the error will decrease and neighborhood path of A and B will be asymptotic to their ideal geodesic path. When samples overspread the surface, the errors vanish.

The size of data set has influence on the intrinsic dimensionality DSR can find, and then affects the performance of DSR. To show this, documents share the same topic of "earn" was chosen from "Reuters-21578" collection. The starting data set contains 50 documents. With 50 samples added at each time, a rather large data set containing 2250 samples is got ultimately. Fig. 4a shows RV-dimensionality curves of DSR when 50, 300 and 1000 documents are adopted. In all these cases, elbow points appear and intrinsic dimensionalities are found. The relation of obtained intrinsic dimensionality to size of data set is shown in Fig. 4b. Obviously, size of data set places influence on the intrinsic dimensionality found by DSR. When samples are too few, the structure of the data set cannot be represented properly and a lower dimensionality is usually

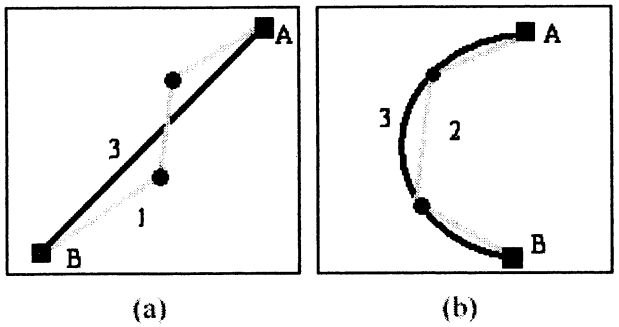


Fig. 3. Two sources of errors in geodesic distance calculation. (a) Part of the unfolded S surface in Fig. 1a. Line 3 is the ideal geodesic path between two arbitrary points A and B, and segments 1 is the path constructed by the neighboring samples in the S surface. Departure of segments 1 from line 3 causes the first kind of error. (b) Cutaway view of the ideal geodesic path. Curve 3 is identical to line 3 in (a), and segments 2 is the path along the Euclidean distance between neighboring samples. Difference between segments 2 and curve 3 results in the second kind of error.

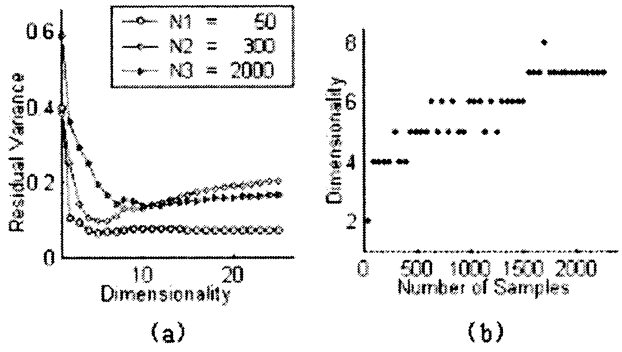


Fig. 4. Influence of data size to intrinsic dimensionality found by DSR. (a) RV-dimensionality curves of DSR when 50, 300 and 1000 documents are adopted. (b) The intrinsic dimensionality retrieved by DSR on data sets with different numbers of documents. Numbers of documents are from 50 to 2250 at interval of 50. The critical point is about 1550 here.

obtained. When number of samples increases, the intrinsic structure is more and more subtly represented. During this process, the dimensionality found by DSR tends to increase gradually. But when the number reaches a critical point, additional samples will not affect the obtained dimensionality any longer since the intrinsic structure has been described sufficiently. The dimensionality we finally arrive at is the true intrinsic dimensionality where the intrinsic structure resides in.

B. Can DSR be applied in other problems?

As we have mentioned, DSR in information recommendation lies in the transferences in evaluation mechanism, which results in the structure of data.

In the experiment detailed above, a document can be expressed by the frequencies of its keywords, in Section II, we deem documents as objects, and vectors of keywords' frequencies as evaluations to the documents. Different documents become similar because of common keywords, which can lead to transference in semantic sense and form the structure of data set.

Then we can find the features of the problems that can be solved by DSR:

1. Objects in the problems can be represented by VSM.
2. Similarities can be defined as Eq. (2).
3. There are transferences of similarities.

So DSR not only can be applied in information recommendation, but also in other fields of information processing, such as information retrieval and text classification. Further more, so long as there is transference of information evaluation among objects, the problem can be solved in similar idea as DSR.

IV. Conclusion

Based on a technique Dissimilarity Reconstruction (DSR) we put forward formerly, the objects in information recommendation are represented as vectors in the intrinsic dimensional space of the data set.

The main idea of DSR is to find the intrinsic structure of the data set by simulating the gradually transferring mechanism in people's evaluation process of information and build relationships between objects according to these transferences.

An experiment is executed on "Reuters-21578" text categorization test collection. Results of experiments show that relationships between objects are well represented by Euclidean distances in the intrinsic dimensional space. So DSR is effective in information recommendation.

Some more discussions are given subsequently, which include influence of data size and potential application of DSR.

By now, representation of objects using DSR has been discussed in detail. Based on this representation, further recommendation process can be done to form a completed recommendation system. Such a recommendation system will be put forward in our future work.

Acknowledgments

We should thank Reuters Ltd. and Carnegie Group for the distribution of text categorization test collection of "Reuters-21578, Distribution 1.0", and thank for Steve Finch and David D. Lewis for the cleanup of the collection. "Reuters-21578, Distribution 1.0" is now available at http://www.daviddlewis.com/resources/testcollections /reuters21578/.

References

[1] Kou Zhongbao, Ban Tao, and Zhang Changshui, "Discovery of Relationships between Interests from Bulletin Board System by Dissimilarity Reconstruction", *The 6th International Conference on Discovery Science*, Hokkaido University Conference Hall, Sapporo, Japan, October 17-19, 2003

[2] M. Balabanovic, "Learning to Surf: Multiagent Systems for Adaptive Web Page Recommendation", Ph.D. dissertation, Stanford University: Department Of Computer Science, 1998.

[3] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, 1983.

[4] J.B. Tenenbaum, V.de. Silvam, and J.C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol.290, pp. 2319–2323, 2000.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Length-Based DNA Computing for 1-Pair Shortest Path Problem

Zuwairie Ibrahim[1], Osamu Ono[2], Yusei Tsuboi[3], and Marzuki Khalid[4]

[1,2,3]Institute of Applied DNA Computing, Meiji University
1-1-1 Higashimita, Tama-ku, Kawasaki, Kanagawa, JAPAN

[4]Centre for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia
Jalan Semarak, 54100 Kuala Lumpur, MALAYSIA

(zuwairie, ono, tsuboi)@isc.meiji.ac.jp, marzuki@utmkl.utm.my

Abstract

The 1-pair shortest path problem of a graph can be solved by exploiting some characteristics of DNA such as length, concentration, and melting temperature. This paper presents an improved length-based DNA computing approach for this problem whereby every path is encoded by oligonucleotides and the path's length is directly proportional to the length of oligonucleotides. Using these properties, gel electrophoresis is performed in order to separate the respective DNA molecules according to their length. It is believed that it is possible to implement the proposed method by laboratory experiments.

Keywords: DNA computing, length-based computation, shortest path problem

1 Introduction

At first, Adleman [1] pioneered the molecular computation by solving seven nodes of Hamiltonian Path Problem (HPP). While this solution was very promising, the problem did not involve shortest path computation at all.

Shortest path HPP is solved by Narayanan and Zorbalas [2] by DNA algorithms. A constant increase of DNA strands according of the actual length of distance is occupied. A drawback of this method is that for some concatenations, concatenated DNA strand of two distances, while less numerically than the longest distance, may be longer than the DNA strands for that longest distance. This may leads to errors in computing shortest path. This scheme however was not been realized by laboratory experiments.

Yamamoto et al. [3] presented concentration-controlled DNA computing for accomplishing a local search for shortest path problem. Although DNA computing with concentration control method enables local search among all the candidate solutions, it is not guaranteed that the most intensive band is the DNA representing the shortest path in the given graph. In addition, it is technically difficult to extract a single optimal solution from the most intensive band. J. Y. Lee et. al [4] proposed temperature gradient-based DNA computing for shortest path TSP problem. They introduced melting temperature (T_m) for this purpose. At certain temperature, DNA strands of correct solutions will be denatured at that temperature and be amplified by Polymerase Chain Reaction (PCR) process. As the denaturation temperature increases, other DNA strands also will be amplified. But the amount of correct solutions will be exponentially increased cycle by cycle, and occupy the major part of the solution.

It is found that the length-based DNA computing has not been implemented in the laboratory experiments yet. Thus, in this paper, we improved the encoding style proposed previously in [2] in order to compute the shortest path by DNA computing. In the proposed method, the edge weights are handled by DNA strands and the actual weights are approximated by the directly proportional length DNA strands. Hence, the important information during the computation is the length of the DNA.

2 Problem formulation

The input to this problem is a weighted, undirected graph $G = (V, E, \omega)$ as depicted in Figure 1. The output of the 1-pair shortest path problem is the shortest (u, v) path and its weight. In the case as shown in Figure 1, if u is V_1 and v is V_5, the weight for the shortest path is given as 27.

3 Computing with DNA

The overall computation described in this section consists of five steps altogether. Step 1 through step 3 is crucial for the generating initial pool, random route

formation, whereas step 4 through step 5 is designed for the computation process and applied to the initial pool of solution.

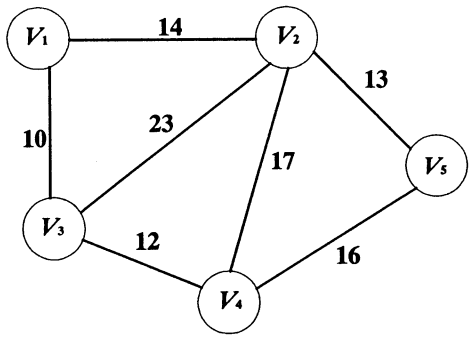


Figure 1. A weighted undirected graph $G = (V, E)$.

Step 1: Let V be the total number of nodes in the graph and E the total number of paths in the graph. The random sequence strands correspond to all nodes (and its complements) and distances (and its complements) are created. Let O_i $(i = 1,\ldots..,\ V)$ and $\overline{O}\_i$ $(i = 1,\ldots..,\ V)$ be the fixed length random sequences correspond to all nodes in the graph and its complements respectively. In the same way, let O_d $(d = 1,\ldots..,\ E)$ and $\overline{O}\_d$ $(d = 1,\ldots..,\ E)$ be the variable length random sequences correspond to all the distances and its complements respectively. The length l of O_d and $\overline{O}\_d$ will be directly proportional to the weight distances. For instance, let say, for a constant proportional increase of 2, all the sequences are placed in Table 1 and Table 2. At the end of this step, oligonucleotides $\overline{O}\_d$ and $\overline{O}\_i$ are synthesized.

Table 1. Fixed Length Node Random Sequences

Node	DNA Code O_i (5' → 3')	Complement $\overline{O}\_i$ (3' → 5')
V_1	TATT	ATAA
V_2	GCGG	CGCC
V_3	GTCG	CAGC
V_4	AGAA	TCTT
V_5	TCCG	AGGC

Step 2: Let i be the start node of a path, d the path's distance and j the end node of that path. Oligonucleotides representing every path between two nodes in the graph are synthesized as follows:

synthesize oligonucleotides
O_i-d-j_end as **HR** *O_i* + **ALL** *O_d* + **ALL** *O_j*
O_j-d-i_end as **HR** *O_j* + **ALL** *O_d* + **ALL** *O_i*
O_i-d-j as **HR** *O_i* + **ALL** *O_d* + **HL** *O_j*
O_j-d-i as **HR** *O_j* + **ALL** *O_d* + **HL** *O_i*

where '**ALL**' represents the whole DNA strand, '**HL**' the left half part, '**HR**' the right half part and '+' is a join. For each path, connecting two identical nodes, each of the nodes could be a starting node, an end node or a connective node in a path. Thus, four identical oligonucleotides will be created for each edge.

Table 2. Variable Length Distance Random Sequences

Distance	DNA Code O_d (5' → 3')	Complement $\overline{O}\_d$ (3' → 5')
D {10}	20	$\overline{20}$
D {12}	24	$\overline{24}$
D {13}	26	$\overline{26}$
D {14}	28	$\overline{28}$
D {16}	32	$\overline{32}$
D {17}	34	$\overline{34}$
D {23}	46	$\overline{46}$

As such, if O_i = **TATT** then $\overline{O}\_i$ = **ATAA**; if O_j = **GCGG** then $\overline{O}\_j$ = **CGCC** and if O_d = **ACGACGTT** then $\overline{O}\_d$ = **TCGTGCAA**. Then the path *O_i-d-j_end* is **TTACGACGTTGCGG** (HR *O_i* + ALL *O_d* + ALL *O_j*), the path *O_j-d-i_end* is **GGACGACGTTTATT** (HR *O_j* + ALL *O_d* + ALL *O_i*), the path *O_i-d-j* is **TTACGACGTTGC** (HR *O_i* + ALL *O_d* + HL *O_j*), and lastly the path *O_j-d-i* is **GGACGACGTTTA** (HR *O_j* + ALL *O_d* + HL *O_i*).

At the end of this step, 28 identical edge strands will be created by a combination of the nodes sequences and distance sequences. Because of this combination, the edges strands *O_i-d-j*, *O_i-d-j_end*, *O_j-d-i*, and *O_j-d- i_end* listed in Table 3 are influenced by the additional nodes bases and hence, unable to handle the weight of each edges. The additional bases would cause errors during the computation because the additional bases composed in the edge sequences reflect the weight value of edges. To compensate the additional bases, the strands corresponds to distance sequences are shorten according to the additional bases and the node sequences in the edge strands are left unchanged. All the modified oligonucleotides are listed in Table 4. This modification however, decreases the

generality of the approach because the weight of edges less than 3 is not allowed.

Step 3: All modified oligonucleotides *O_i-d-j(modified)*, *O_i-d-j_end(modified)*, *O_j-d-i(modified)*, and *O_j-d-i_end(modified)* followed by all complement oligonucleotides ($\overline{O}\_i$ and $\overline{O}\_d$) are inserted into a test tube as shown in Figure 2. Then DNA ligase reaction is performed in which random routes through the graph are formed. In this way, the initial solution, which contains a large set of random routes of the graph are generated. After that, in order to make sure that no sticky end strands exist in the solution, polymerase enzyme is used for the extension where the sticky end strands are altered to become blunt end strands.

Table 3. Edge Sequences

Edges	DNA Code (5' → 3')
$V_3 - V_1$	CG 20 TATT TT 20 GTCG CG 20 TA TT 20 GT
$V_4 - V_3$	AA 24 GTCG CG 24 AGAA AA 24 GT CG 24 AG
$V_2 - V_5$	GG 26 TCCG CG 26 GCGG GG 26 TC CG 26 GC
$V_1 - V_2$	TT 28 GCGG GG 28 TATT TT 28 GC GG 28 TA
$V_5 - V_4$	CG 32 AGAA AA 32 TCCG CG 32 AG AA 32 TC
$V_4 - V_2$	AA 34 GCGG GG 34 AGAA AA 34 GC GG 34 AG
$V_2 - V_3$	GG 46 GTCG CG 46 GCGG GG 46 GT CG 46 GC

At this moment, an initial solution is already obtained and it is time to filter out the right routes among the vast random routes of the graph. Unlike the conventional filtering, it is not merely throwing away the unwanted strands but rather copying the right strands exponentially by incredibly sensitive PCR process.

Step 4: Amplify the DNA molecules that contain the start node V_1 and end node V_5. The amplification is executed by applying PCR operation to respective DNA molecules with primer AGGC and primer TT, which is the complement of V_5 and 3' half of V_1 respectively.

Table 4. Modified Edge Sequences

Edges	DNA Code (5' → 3')
$V_3 - V_1$	CG 14 TATT TT 14 GTCG CG 16 TA TT 16 GT
$V_4 - V_3$	AA 18 GTCG CG 18 AGAA AA 20 GT CG 20 AG
$V_2 - V_5$	GG 20 TCCG CG 20 GCGG GG 22 TC CG 22 GC
$V_1 - V_2$	TT 22 GCGG GG 22 TATT TT 24 GC GG 24 TA
$V_5 - V_4$	CG 26 AGAA AA 26 TCCG CG 28 AG AA 28 TC
$V_4 - V_2$	AA 28 GCGG GG 28 AGAA AA 30 GC GG 30 AG
$V_2 - V_3$	GG 40 GTCG CG 40 GCGG GG 42 GT CG 42 GC

After the PCR operation is accomplish, there should be a numerous number of DNA strands representing the start node V_1 and end node V_5 traveling through a number

possible routes. Four types of DNA strands traveling through four different routes are given in Figure 3.

Step 5: The output solution of PCR operation is brought for gel electrophoresis operation. During this operation, the DNA molecules will be separated in term of its length. This operation is based on the fact that DNA molecules are negatively charged [5]. Hence, by putting them in an electric field, they will move towards the positive electrode at different speed. The longer molecules will remain behind the shorter ones. For better understanding of the gel electrophoresis, an example output of gel electrophoresis is given in Figure 4 [6]. Thus, it is able to separate and differentiate the shortest DNA molecules representing the answer to the 1-pair shortest path problem.

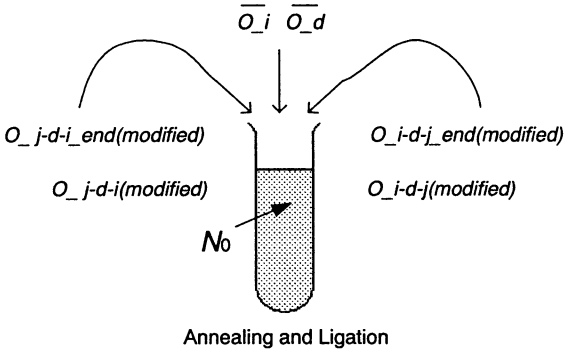


Figure 2: Performing annealing and ligation processes.

TT	24	GCGG	20	TCCG
AA	24	CGCC	20	AGGC

(a)

TT	16	GTCG	20	AGAA	26	TCCG
AA	16	CAGC	20	TCTT	26	AGGC

(b)

TT	24	GCGG	30	AGAA	26	TCCG
AA	24	CGCC	30	TCTT	26	AGGC

(c)

TT	16	GTCG	42	GCGG	30	AGAA	26	TCCG
AA	16	CAGC	42	CGCC	30	TCTT	26	AGGC

(d)

Figure 3: The DNA molecules amplified after the PCR operation (a) DNA strand $V_1 - V_2 - V_5$ (54bp) (b) DNA strand $V_1 - V_3 - V_4 - V_5$ (76bp) (c) DNA strand $V_1 - V_2 - V_4 - V_5$ (94bp) (d) DNA strand $V_1 - V_3 - V_2 - V_4 - V_5$ (132bp)

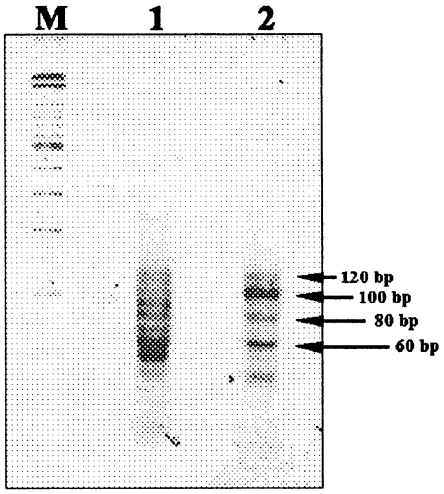


Figure 4. Gel electrophoresis output where lane M is DNA size marker. Lane 1 and 2 are used for the tested DNA molecules.

4 Concluding remarks

This paper discussed an improved length-based DNA computing to solve 1-pair shortest path problem of a graph or network. This method proposed the proportional length of DNA to encode the weight of each edge. The implementation by laboratory experiments will be carried out soon. It is expected that the approach can be used to solve another graph problems based on DNA computing as a medium for computation.

References

[1] Adleman L (1994), Molecular computation of solutions to combinatorial problems. Science 266: 1021-1024

[2] Narayanan A, Zorbalas S (1998), DNA algorithms for computing shortest paths. Proceedings of genetic programming, 1998, pp. 718-723

[3] Yamamoto Y, Kameda A, Matsuura N, Shiba T, Kawazoe Y, Ahochi A (2002), Local search by concentration-controlled DNA computing. International journal of computational intelligence and applications 2: 447-455

[4] Lee JY, Shin SY, Augh SJ, Park TH, Zhang BT (2003), Temperature gradient-based DNA computing for graph problems with weighted edges. Lecture notes in computer science 2568:73-84

[5] Paun G, Rozenberg G, Salooma A (1998) DNA computing: New computing paradigms. Springer-Verlag, Heidelberg

[6] Yamamoto M, Kameda A, Matsuura N, Shiba T, Kawazoe Y, Ahochi A (2002), A separation method for DNA computing based on concentration control. New generation computing 20: 251-262

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

General Protocol for Evaluating the Degree of Occurrence of Mis-Hybridization

Satoshi Kashiwamura
Graduate School of Engineering
Hokkaido University North 13, West 8
Kita-ku, Sapporo 060-8628, Japan
kashiwa@dna-comp.org

Atsushi Kameda
CREST, Japan Science and Technology
Corporation (JST), Honmachi 4-1-8
Kawaguchi 332-0012, Japan
kameda@dna-comp.org

Masahito Yamamoto
PRESTO, Japan Science and Technology
Corporation (JST), Honmachi 4-1-8
Kawaguchi 332-0012, Japan
Graduate School of Engineering
Hokkaido University North 13, West 8
Kita-ku, Sapporo 060-8628, Japan
masahito@dna-comp.org

Azuma Ohuchi
CREST, Japan Science and Technology
Corporation (JST), Honmachi 4-1-8
Kawaguchi 332-0012, Japan
Graduate School of Engineering
Hokkaido University North 13, West 8
Kita-ku, Sapporo 060-8628, Japan
ohuchi@dna-comp.org

Abstract

In the field of DNA Computing, DNA sequences that do not produce erroneous mis-hybridizations are needed. There are a lot of works discussing how to design the set of such DNA sequences in computer simulation but there are few works that evaluate the performance of designed sequences in chemical experiments.

In this paper, we propose a systematic protocol for evaluating the error-excluding capability of DNA sequences. In addition, we report the result of DNA sequences designed by the *template method* according to our proposed protocol.

keywords : DNA computing, DNA Sequence Design, mis-hybridization, sequence evaluation, standard protocol.

1 Introduction

DNA (Deoxyribo Nucleic Acid) has a structure that can save an enormous amount of data and the ability to perform massive parallel reactions. With a view toward utilizing such an attractive feature for computation, DNA computation has begun (Adleman[1], Lipton[2]). To implement a successful DNA computation, the hybridizations should occur as designed. Otherwise, unplanned hybridization (i.e. mis-hybridization) can occur, which prevents correct results. Therefore, a first step in DNA computation is to find sequences that minimize mis-hybridization, and this has been termed the DNA sequence design problem. In this paper, we call the degree to avoid mis-hybridization "specificity", and the sequence that can avoid mis-hybridization is the "specific sequence". Today, there are many works to discuss how to find the set of specific sequences (Arita and Nishikawa et al.[3], Garzon and Deaton et al.[4], Frutos and Liu at al.[5] etc). The main approach for designing specific sequences is to arrange the constraints to avoid any unplanned reactions, and then find a set of DNA sequences, in which all sequences satisfy all given constraints. However, there is a serious problem in this approach: it is not known how effective the constraints are for avoiding mis-hybridizations because detailed knowledge of hybridization of DNA is still inadequate. That is, it is still not clear whether a designed sequence can definitely avoid mi-hybridization. Despite this, most works have not checked the performance of designed sequences through chemical experiments. Based on such background, we propose a systematic protocol for evaluating the specificity of designed DNA sequences in this paper. Our final goal is to establish the constraints or sequence design strategies so that we can obtain perfect sequences with absolute specificity. This work is the first step. In addition, we evaluate the specificity of DNA sequences designed by the *template method* in accordance with our proposed

protocol.

2 Basis of DNA and Experimental tools

A DNA sequence is a string structure consisting of four nucleotide bases (A: adenine, C: cytosine, G: guanine, and T: thymine) and in one direction (from the 5' end to the 3' end). A complementary sequence is obtained by substituting base A with T and base C with G, and vice versa, and reversing its direction. For example, the complementary sequence of 5'-ATGGCTAC-3' is 5'-GTAGCCAT-3'. A DNA sequence hybridizes to its complementary sequence and forms double-stranded DNA due to a hydrogen bond between A and T or between G and C. This is the main engine of DNA computing. The complementary sequence of a DNA sequence x is denoted by $\overline{x}$. PCR (Polymerase Chain Reaction) is a popular technique for amplifying target domain of DNA between particular segments where short DNAs, called primers, hybridize. PAGE (Poly-Acrylamide Gel Electrophoresis) is a technique for separating DNAs based on their length, where DNA is visualized as bands on gel and their position corresponds to its length. The intensity of the band represents the amount of DNA.

3 Standard protocol for evaluating specificity

Here, we describe the standard protocol of chemical experiments to evaluate the specificity of sequences. Our proposal protocol also evaluates the specificity for the area where arbitrary sequences are concatenated. Arita call the constraint "Comma-Free Constraint" (Arita[6]). Satisfying the comma-free constraint allows for a flexible application in DNA computing and contributes to the design of a lot of sequences with rather high specificity. We consider comma-free constraint to be very important; therefore, our evaluation protocol is intended for sequences that satisfy comma-free constraint.

3.1 How to measure specificity?

To achieve our purpose, we must first of all define how to evaluate specificity. Specificity depends strongly on the reaction temperature. When the reaction temperature is too high, it is difficult for even correct hybridization to occur. On the other hand, if the reaction temperature is too low, mis-hybridization also tends to occur. That is, when the range of reaction temperature is wider such that target reaction is produced and the non-target is eliminated, the sequence has the good specificity. In this work, we define the sequence specificity as the temperature range where only good reactions can occur.

3.2 Standard protocol

1. Prepare a set of the specific sequence (length L) satisfying comma-free constraints.
2. Select N sequences from the set at random.
3. Concatenate selected N sequences in random order. Call the long sequence (length: $L \times N$) made of selected N sequences the *Basic*. Each sequence area in the *Basic* is called a block. (see Fig. 1)
4. Select a primer pair so that m block (m is constant) may be amplified, and perform PCR with the primer pair. Here PCR is performed by various annealing temperatures from low to high. (see Fig. 2)
5. Determine both the amount of target PCR products (size of m blocks) and the total amount of other products. The amount of target product corresponds to the degree of correct hybridizations, on the other hand, that of the others corresponds to mis-hybridization. However, in this method, there is some un-detectable mis-hybridization directly, as shown in Fig. 4. We therefore estimate the amount of such un-detectable mis-hybridization by investigating the amount of the target PCR product. In case when the amount of target product is small, the primer will tend to hybridize the wrong area, and as a result it will be difficult for correct hybridization to occur. When the total amounts of the non-target products are low and that of the target product is high, the sequences are good. Section 5 provides details of how to analyze the specificity from the data.

3.3 Merits of this protocol

1. Detailed data
   First of all, this protocol includes comma-free constraints. There has been no report on evaluating the sequence specificity with comma-free constraints. Moreover, by chaining the reaction temperature, we can obtain more exact data.

2. Large amount of data
   Once the *Basic* is created, we can obtain data about the real specificity by only changing the primer pair. Therefore, a large amount of data can be obtained very easily.

3. Low cost and convenience
   The number of required DNA is small, compared to the amount of obtainable data from them. Furthermore, no special modification for the DNA is needed. Therefore, the cost of this chemical experiment is very low. In addition, the only necessary apparatus for this experiment is a thermal cycler, so no special apparatus is not needed.

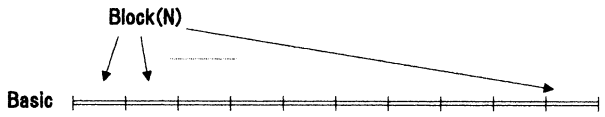


Figure 1: *Basic* and blocks

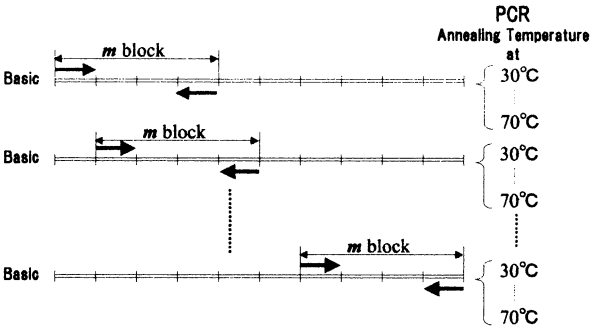


Figure 2: Amplification of m block by PCR at various annealing temperatures

4 Chemical experiments

In this section, we try to collect data to evaluate the specificity of some sequences by our proposed protocol. Here, we deal with sequences designed by the *template method* (Arita and Kobayashi[7]). We selected the template method from among other design algorithms for the following three reasons. First, the sequences designed by *template method* satisfy comma-free constraint. Second, the tightness of the constraints each

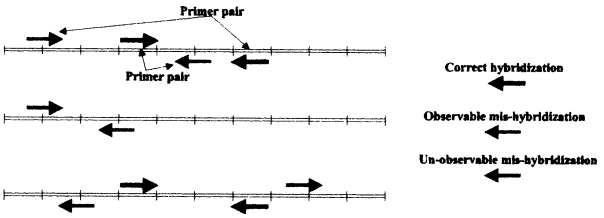


Figure 4: Detectable and un-detectable mis-hybridization

sequence satisfies is even. Third, there is no report of chemical experiments using sequences designed by the *template method.*

The outline of our chemical experiment is described as follows.

1. We prepared a set of sequences using the *template method.* We set length L to 16 and selected 1110001101000100 from the GC template database as a template. We applied the Nordstrom-Robinson code to this template, and obtained 64 sequences are obtained.

2. We selected 20 sequences randomly.

3. We created the *Basic* (320 bp). The 20 sequences constituting the *Basic* were labeled 0, 1, 2 ... 19 from the 5' end to the 3' end. The base sequences and detailed process to construct *Basic* are shown at http://harmony.complex.eng.hokudai.ac.jp/DNA-comp.

4. We selected the primer pair so that five blocks (80 bp) could be amplified. The range of the annealing temperature was from 30 to 70° C, PCR was performed for 20 cycles at 5° C intervals. We chose 20 cycles because the pre-experiments showed that 20 cycles was just before where the plateau was reached in almost all cases (data not shown). We then ran all samples by each primer pair and each annealing temperature on PAGE. The details are also described in the above web site. As instance, one result of PAGE is shown in Fig. 5.

5. We determined the quantity of target product and non-target products of all samples by calculating the intensity of the 80-bp band and that of the bands of other lengths, respectively. Each intensity was changed to relative intensity based on the

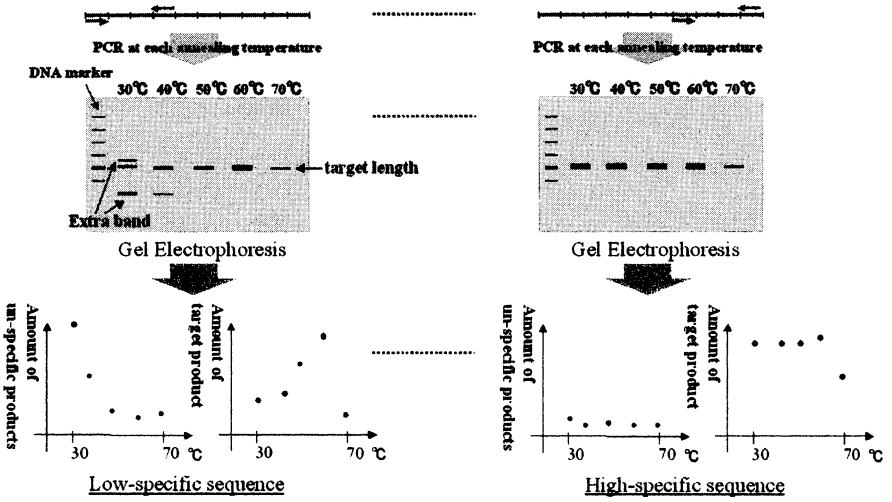


Figure 3: Method of evaluating specificity

intensity of the 100-bp band in the 100-bp DNA marker. We provide an analysis of the obtained data is explained in the next section.

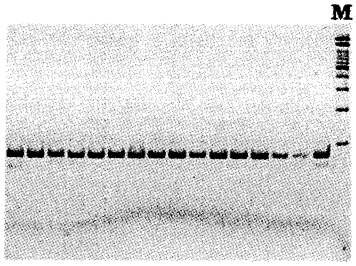


Figure 5: This figure is the result of PAGE when the annealing temperature is 40° C. M: 100-bp DNA. Each lane from left to right shows the PCR samples by primer pair $\{0,\overline{4}\}$, $\{1,\overline{5}\}$, $\{2,\overline{6}\}$ ... $\{15,\overline{19}\}$ at 40° C, respectively.

5 Analysis data and discussion

Here, we introduce two analyses. One is useful for investigating the specificity of individual sequences. The other is useful for observing the properties of a whole set of sequences.

In the first analysis, the obtained data is plotted on a two-dimensional graph in which the x axis indicates the reaction temperature and the y axis indicates the intensity of the target and non-targets (Fig. 6). This arrangement indicates the relation between the yield of the target or non-targets and the annealing temperature change. The points plotted by the square show the relative intensity of the target at each annealing temperature and the points plotted by the regular triangle show that of the non-targets. The scale on the left side of the graphs is for the target and that on the right side is for the non-targets.

It is clear from these graphs that the values of the non-targets become smaller as the annealing temperature increases, and this feature is a reasonable result. Moreover, we can also see that the intensity of the target is the most high around Tm, from 55° C to 60° C. In other words, when the annealing temperature is low, the intensity of the target is also low. From this fact, we can expect that the primer has caused not only correct hybridization, but also many unwanted reactions, including mis-hybridization, simultaneously. Therefore, the probability that primers act correctly falls, and as a result, the target yield decreases. The fact that the target intensity becomes comparatively small when the intensity of the non-targets is high supports our prediction.

Focusing on each sequence, we can clearly observe a variety of features of the sequences. Some sequences ($\{0, \overline{4}\}$ or $\{6, \overline{10}\}$) remove non-targets at low temperature. On the other hand, some sequences produce mis-

hybridization even at rather high temperatures such as ($\{2, \overline{6}\}$ or $\{15, \overline{19}\}$), and some sequences are difficult to hybridize correctly at any annealing temperature. From these results, we think $\{4, \overline{8}\}$, $\{10, \overline{14}\}$ and so on, are good sequences. On the other hand, $\{2, \overline{6}\}$ and $\{15, \overline{19}\}$ may not be good.

The second analysis plots the data in a scatter plot, which expresses the relation between the intensity of target and non-targets for each temperature (Fig. 7). The x axis or the y axis is the intensity of the target or the non-target, respectively. If plotted on the lower right, all sequences have high specificity. If many points are plotted on the upper left, it is low. We can observe that the points are widely scattered at low temperature, then the points move to the lower right gradually with a rise in temperature. When the temperature passes over Tm, the points on the graphs move to the lower left rapidly. The merit of this graph is that we can evaluate the whole set of sequences at a glance.

6 Summary

This paper proposes a general protocol for evaluating the specificity of sequences designed under comma-free constraints through a chemical experiment. We tried to evaluate the specificity of sequences designed by the template method. Results showed that even if all sequences satisfy the constraints evenly, the specificity evaluated by the chemical experiments is not uniform. That is, it turns out that there is a gap between satisfying constraints and the specificity in reaction tube. Our final purpose is to bridge this gap.

Using our proposed protocol, experimental data from chemical reaction is obtained easily and its cost is low. Our proposed protocol has the capacity to evaluate the comma-free constraints, and these data are effective in case various sequence design strategies are compared.

In this paper, we selected the *template method* from among algorithms to design specific sequences. From the experimental results, because of the small number of data and no data for comparison, we cannot say whether the *template method* is the best algorithm. However we felt that the sequences used here effectively eliminate the non-target product at such low temperatures.

This research is just the first stage. In future work, we will try to evaluate the specificity of sequences designed by other methods, investigate the cause of that gap and propose new constraints by which we can estimate specificity more exactly.

Acknowledgements

The authors would like to thank M. Arita of the University of Tokyo in Japan for helpful advice and discussions, and Y. Kita of the University of Hokkaido in Japan for help during preparation of the figures.

References

[1] Adleman, L.M. (1994), Molecular Computation of Solutions to Combinatorial Problems, Science 266: 021-1024.

[2] Lipton, R. (1995), DNA Solution of Hard Combinatorial Problems, Science 268: 542-545.

[3] Arita, M., Nishikawa, A., Hagiya, H., Komiya, K., Gouzu, H. and Sakamoto, K. (2000) Improving Sequence Design for DNA Computing, Proc. of GECCO'00 (Genetic and Evolutionary Computation Conference): pp.875-882.

[4] Garzon, M., Deaton, R., Nino, L.F., Stevens, E. (1998), Encoding Genomes for DNA Computing, Proc. of the Third Annual Genetic Programming Conf: 684-690.

[5] Frutos, A.G., Liu, Q., Thiel, A.J., Sanner, A.M.W., Conndon, A.E., Smith, L.M. and Corn, R.M. (1997), Demonstration of A Word Design Strategy for DNA Computing on Surface, Nucleic Acids Research, Vol.25, No.23: 4748-4745.

[6] Arita, M. (2002), DNA Sequence Design Using Multiple Template, Preliminary Proceeding of 9th International Meeting on DNA Based Computers, Madison, Wisconsin, June 2003: 204.

[7] Arita, M., Kobayashi, S. (2003), DNA Sequence Design Using Templates, New Generation Computing 20: 263-277.

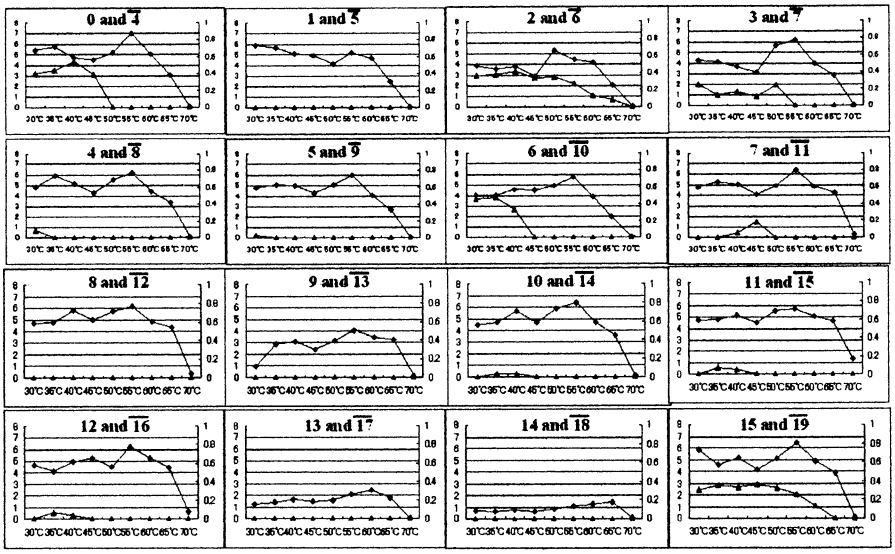


Figure 6: The graphs show the relation between the intensity of the target or non-target and temperature change. The x axis is temperature, and the y axis indicates the intensity of the target and non-targets. The points plotted by the square show the relative intensity of the target (left side scale) and plotted by the regular triangle show that of the non-targets (right side scale).

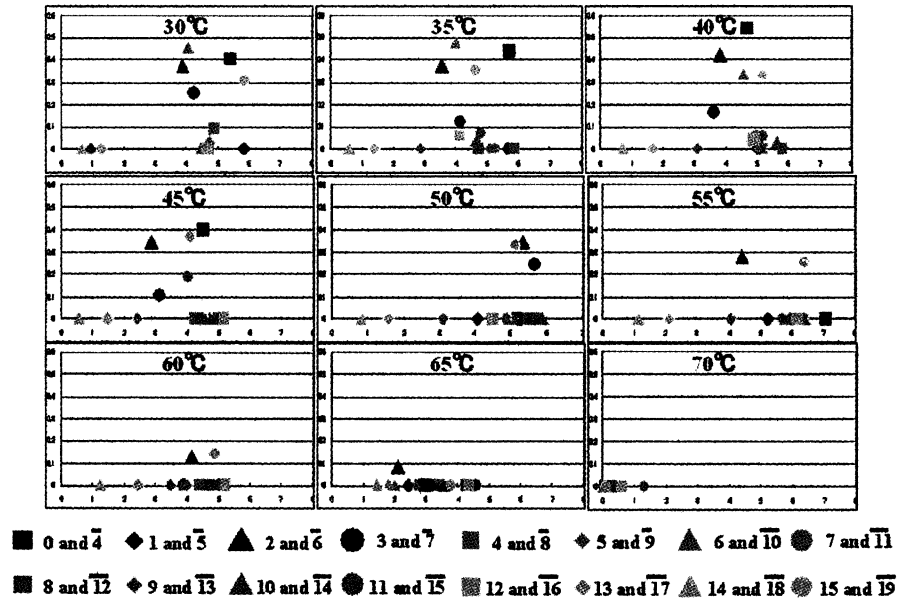


Figure 7: These scatter plots show the relation between the intensity of the target and that of the non-target for each annealing temperature. The x axis is the intensity of the target and the y axis is that of the non-target. Each result is plotted corresponding to the intensity of its target and non-target under each annealing temperature.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A Simple Model for the Evolution of a Lexicon

Edgar E. Vallejo*
Computer Science Department
Tecnológico de Monterrey
Campus Estado de México
Atizapán, Estado de México 52926

Charles E. Taylor†
Department of Organismic Biology
Ecology and Evolution
University of California Los Angeles
Los Angeles, CA 90024

Abstract

This paper explores the evolution of communication in a community of agents. Experimental results show that agents are capable of evolving a shared lexicon describing robot behavior. Categorization of perceptions arises as an emergent property of the imitative interaction of agents.

1 introduction

The origin and evolution of language is an excellent domain for studying fundamental questions of artificial life research. Previous work by Arita[1], Steels[2], Hasimoto and Ikegami[3], and Kirby[4], among others, have shown that we are able to explore important issues such as emergence, self-organization and cultural evolution within this framework.

The emergence of symbolic communication is one of the most significant transitions in the evolution of language and at the core of what is desired for adaptive robotics. This implies the ability to acquire concepts, to ground symbols into concepts, and to propagate those symbols in a community of other agents

Previous work on evolving symbolic communication systems was largely based on the approach proposed by Hurford[5]. The core of this model is a pair of lexical matrices in which a fixed collection of symbols and meanings are probabilistically correlated. An underlying one-to-one correspondence between symbols and meanings is assumed. The categorization and generalization capabilities of agents are limited to a few cases of synonymy and homonymy. Most work based on lexical matrices have focused on the evolutionary behavior of communicative strategies [6][7]. We believe that for a symbolic communication system to properly work it must adequately capture patterns of categorization.

McLennan [8] has demostrated that finite state machines (FSMs) can be used to evolve a highly coordinated symbolic communication system. This holds much promise. For example, Lee et al[9] found that FSMs can be used as appropriate models for symbol grounding. Moreover, these models can readily discriminate a potentially infinite collection of inputs.

This paper extendes the study of Lee et al [9] from a single agent to the emergence of symbolic communication in a population of agents. In our model, agents communicate by producing an utterance in response to a sensory stimulation. The communicative behavior of agents is produced by finite state transducers. We assume that communication is an evolutionary behavior: agents who communicate well produce more offspring. Agents achieve lexicon formation by imitation: the communicative success of an agent depends on his imitative ability.

Experimental results show that our model is capable of producing an emergent lexicon describing robot behavior in a population of agents. Furthermore, we show that meaningful categories of robot behaviors emerge in our model without any explicit selective pressure for categorization.

2 The model

The experiments were conducted by using the methodology proposed by Steels[10], inspired by the language game of Wittgenstein[11]. Under this framework, an experiment in artificial language evolution always has the following ingredients: (1) an interaction protocol for the agents, (2) an agent architecture, (3) an environment, and (4) a set of measures for success in communication.

2.1 Interaction protocol

The model considers a population of communicating agents. During the simulation, an agent interacts with a subpopulation of randomly selected agents by

*vallejo@itesm.mx
†taylor@biology.ucla.edu

using a simple pairwise imitation game: the agent attempts to match the utterance produced by other agents in response to a particular sensory stimulation.

2.2 Agent architecture

The communicative behavior of agents is produced by finite state transducers (FSTs) [12]. An extension of the Mealy machine model was formulated for this purpose as follows. A *one-symbol output Mealy machine* is a 6-tuple $(Q, \Sigma, \Gamma, \delta, \phi, q_0)$, where

1. Q is a finite set of states,
2. Σ is the input alphabet,
3. Γ is the output alphabet,
4. $\delta : Q \times \Sigma \rightarrow Q$ is the transition function,
5. $\phi : Q \times \Sigma \rightarrow \Gamma$ is the output function,
6. $q_0 \in Q$ is the start state.

Let $M = (Q, \Sigma, \Gamma, \delta, \phi, q_0)$ be a one-symbol output Mealy machine and $w = w_1 w_2 \cdots w_n$ be a string over the input alphabet Σ. The machine M produces the output symbol u from the output alphabet Γ in response to input w, $M(w) = u$, if a sequence of states $r_0, r_1, \ldots, r_n$ exists in Q with the following conditions:

1. $r_0 = q_0$
2. $\delta(r_i, w_{i+1}) = r_{i+1}$ for $i = 0, \ldots, n-1$, and
3. $\phi(r_{n-1}, w_n) = u$

The output symbol u in $M(w) = u$ is viewed as the utterance that is produced by the agent described by M to verbalize the situation encoded by the input string w. The lexicon of a population of agents P for a set of perceptions W, $L(P, W)$, is defined as the collection of all different output symbols $u \in \Gamma$ produced by all agents $a \in P$, in response to all inputs $w \in W$.

At each step of the simulation, two agents are selected according to their communicative efficiency. These agents produce a new offspring by means of genetic operators. One-point recombination and point mutation operate on a linear representation of agents. M is represented as linearly by:

$$q_{0,0}u_{0,0}q_{0,1}u_{0,1}\cdots q_{1,0}u_{1,0}q_{1,1}u_{1,1}\cdots q_{|Q|,|\Sigma|}u_{|Q|,|\Sigma|}$$

where $q_{i,j} \in Q$ indicates the table entry at row i and column j of the tabular version of δ, $u_{i,j} \in \Gamma$ indicates the table entry at row i and column j of the tabular version of ϕ. $|Q|$ and $|\Sigma|$ are the cardinalities of the set of states and the input alphabet, respectively.

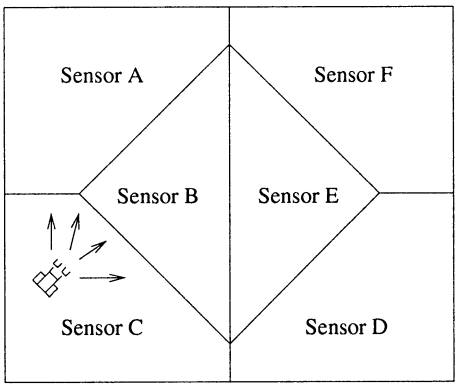


Figure 1: A robot wandering around a room

Behavior	Sensor array data
Wall following 1	CDFACDFACDFA
Wall following 2	CDFACDFACDFACDFA
Random walking 1	AEFFFDDFCAADBAAAE
	CCCCCCCCCCEACCAEC
	ADDEC
Random walking 2	CCAAAABBCFFEAADFE
	ACFBBFEACEDDDDDFF
	FFFFFFFFFFFBCFFEA
	ADDEFDECCEECCCAEC

Table 1: Robot movement data set

2.3 Environment

The proposed model was used explore the evolution of a lexicon describing robot behavior. Figure 1 shows a robot wandering in a room containing a distributed array of sensors. A robot activates the sensor in the area it occupies. Table 1 shows a data set that describes the movement of a robot at discrete time steps, kindly provided to us by Richard Brooks and David Friedlander. (See [13] for further details).

2.4 Measure of communicative success

The communicative success of agents was defined as the ability to imitate the communicative behavior of other agents. Let $a \in P$ be an agent and $S \subseteq P$ a non empty collection of agents. The communicative efficiency of a with respect to S and input string w, $E(a, S, w)$, is defined as

$$E(a, S, w) = \frac{\sum_{s \in S} e(a, s, w)}{|S|}$$

where $e(a, s, w) = 1$ if a and s produce the same

Parameter	Value
Simulation steps	1000–2000
Number of agents in P	128–512
Number of agents in S	4–32
Number of states in Q	4–16
Number of symbols in Σ	6
Number of symbols in Γ	4–16
Recombination probability p_r	0.6–0.7
Mutation probability p_m	0.001–0.01

Table 2: Parameters for simulations

output symbol u in response to input w, and 0 otherwise; $|S|$ is the cardinality of S. The generalization of this measure to a collection of perceptions W, $E(a, S, W)$, is straightforward.

Experiments were conducted to investigate whether a population of imitative agents perceiving the above environment is likely to arrive to a shared lexicon describing robot behavior. The conducted simulations are described by the following algorithm:

1. Create an initial random population P of agents
2. `Do until` number simulation steps N is met
   (a) `For each` agent $a \in P$ `do`
      i. Select a subpopulation $S \subseteq P$ of agents at random
      ii. Measure the imitative success $E(a, S, W)$ for a set of perceptions W

   `End for`
   (b) Select two individuals $a_1 \in P$ and $a_2 \in P$ for reproduction based on their communicative efficiency
   (c) Produce an offspring a_{new} from a_1 and a_2 using one-point recombination and point mutation, with probabilities p_r and p_m, respectively
   (d) Select a random individual $a_{old} \in P$
   (e) Replace a_{old} by a_{new}

`End do`

3 Results

Several simulations were conducted using different combinations of parameter values as shown in table 3. The following were the major results:

1. Agents arrived to a shared lexicon describing robot behavior. However, simulations showed that there exists a threshold condition on the number of interactions required to achieve convergence in communication. Figure 2 shows the results of the simulations for different sizes of S at the proximity of the threshold value.

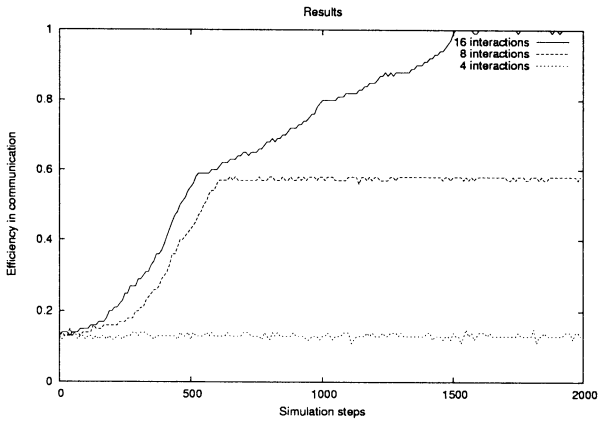


Figure 2: Simulation results

Behavior	Output
Wall following 1	D
Wall following 2	D
Random walking 1	B
Random walking 2	B

Table 3: Results

2. The same communicative behavior of the agents was achieved by means of several different machines. In general, results showed that machine convergence is not a necessary condition for convergence in communication.

3. Agents produced a meaningful emergent categorization of robot behaviors. Moreover, agents produce a meaningful generalization of perceptions when providing them with additional inputs. Table 2 shows the lexicon evolved in a typical simulation.

4. The communicative behavior of agents is produced by compact machines. In general, agents prefer machines with fewer states than the maximum number of states allowed in the simulations.

4 Discussion

The overall results indicate that given a sufficient number of interactions, a community of imitating agents is capable of evolving a shared lexicon describing robot behavior. Surprisingly, a meaningful categorization of perceptions emerges as a side effect of

the imitative interaction of agents. Moreover, the use of FSTs allow agents to perceive a potentially infinite number of inputs and to capture regularities in perceptions.

4.1 Why imitation works

From the evolutionary perspective, how good is imitation for evolving a shared lexicon? Could an alternative communicative strategy invade a population of imitating agents?

Maynard-Smith[14] has demostrated that game theory can be used as a framework to explain the evolution of most phenotypic traits in situations in which fitness of a trait depend on what others are doing. He has also provided the notion of evolutionary stable strategy (ESS). An ESS is a phenotype such that, if almost all individuals have that phenotype, no alternative phenotype can invade the population.

In our model, success in imitation depends on the particular utterances produced by others. Furthermore, we have verified that imitation is an ESS in our model. However, further research is required to investigate the conditions under which imitation would fail. Previous studies have provided some insights [15][7].

4.2 Why categorization emerges

In general, experimental results showed that there is an evolutionary preference for agents which underlying FSTs have fewer states. These machines can capture regularities of inputs in an efficient way. Moreover, previous studies on language evolution have suggested a plausible relationship between compression and generalization in language evolution [16].

However, finite state machines can not capture all interesting robot behaviors in our model. In effect, it is theoretically imposible to learn any arbitrary class of languages [17]. Nevertheless, we believe that the consideration of more powerful computational models, (e.g. Push-down automata), will allow us to capture less restricted and more interesting patterns of robot behavior.

Acknowledgements

We thank Richard Brooks and David Friedlander for kindly providing to us their robot data. We also thank Edward Stabler for useful discussion regarded the limitations of formal computational models and their application to the analysis of robot data. This work was supported by the UC MEXUS-CONACYT, the UCLA Center for Embedded Sensor Networks, the Defense Advance Research Projects (DARPA), administered by the Army Research Office under Emergent Surveillance Plexus MURI Award No. DAAD19-01-1-0361, and DARPA MURI administered by the US Air Force No. F49620-01-1-0361. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

References

[1] Arita T, CE Taylor (1996), A simple model for the evolution of communication. In: LJ Fogel, PJ Angeline, T Back, eds, *Evolutionary Programming V.* The MIT Press, pp.405–410.

[2] Steels L (1996). Self-organizing vocabularies. In: C Langton and T Shimohara (eds), *Artificial Life V. Proceedings of the Fifth International Conference on Artificial Life*, The MIT Press, pp.179–184.

[3] Hashimoto T, T Ikegami (1996), Emergence of net-grammar in communicating agents, *BioSystems* 38:1–14.

[4] Kirby S (2000), Syxtax without natural seleccion: How compositionality emerges from vocabulary in a population of learners. In: C Knight, M Studdert-Kennedy, JR Hurford (eds), *Approaches to the Evolution of Language*, Cambridge University Press, pp 303–323.

[5] Hurford JR (1989), Biological evolution of the Saussurean sign as a component of the language acquisition device. *Lingua*, 77(2): 187–222.

[6] Oliphant M, J Batali (1997). Learning and the emergence of coordinated communication. *The Newsletter of the Center for Research in Language*, 11(1).

[7] Nowak MA, JB Plotkin, D Krakauer (1999). The evolutionary language game. *Journal of Theoretical Biology*, 200(2):147–162.

[8] MacLennan B (1992), Synthetic Ethology: An approach to the study of communication. In C Langton, CE Taylor, D Farmer et al, (eds), *Artificial Life II.* Addison-Wesley, pp.631–658.

[9] Lee Y, J Riggle, TC Collier et al (2003), Adaptive communication among collaborative agents: Preliminary results with symbol grounding. In: M Sugisaka, H Tanaka (eds), *Proceedings of the Eighth International Symposium on Artificial Life and Robotics* (AROB8th), Beppu, Oita Japan, Jan 24-26, 2003, pp.149-155.

[10] Steels L (2003) The evolution of communication systems by adaptive agents. In: D Kudenko, E Alonso, D Kazakov (eds) *Adaptive Agents and MultiAgent Systems:Adaptation and Multi-Agent Learning.* LNAI 2636, Springer-Verlag, pp.125–140.

[11] Wittgenstein L (1953), *Philosophical Investigations*, Oxford: Blackwell.

[12] Hopcroft JE, JD Ullman (1979) *Introduction to Automata Theory, Languages and Computation*, Addison Wesley.

[13] Kobele G et al (2004). These proceedings.

[14] Maynard-Smith J (1982), *Evolution and the Theory of Games.* Cambridge University Press.

[15] Ackley DH, ML Littman (1994) Altruism in the Evolution of Communication. In: RA Brooks, P Maes, (eds), *Artificial Life IV. Proceedings of the Fourth International Conference on Artificial Life*, The MIT Press, pp.40–48.

[16] Teal T and CE Taylor (2000), Effects of compression on language evolution. *Artificial Life* 6(2):129–143.

[17] Angluin D. (1980), Inductive inference of formal languages from positive data, *Information and Control* 45:117-135.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Evolution of investment strategy in an environment offered by the real tick data

Mieko Tanaka-Yamawaki and Tomohiro Motoyama

Department of Information and Knowledge Engineering, Tottori University, Tottori, 680-8552 JAPAN
Email: mieko@ike.tottori-u.ac.jp, motoyama@ike.tottori-u.ac.jp

Abstract

We attempt to generate a series of investment strategy that suite the changing environment, by means of evolutionary techniques in a broad sense inspired by Lindgren's model originally proposed to deal with the repeated games of prisoner's dilemma. Specifically we study the problem of predicting the next motion of the tick-wise price fluctuation by using evolutional approach. We regard each strategy as an agent, which evolves to adapt the changing environment offered by the real financial time series. We focus on the directions of tick-wise motion and code them by {0,1,2} corresponding to {Down, Flat, Up}, and assume that a strategy depends only on the history of its past motions. We have obtained the best strategy by training the initially random agents, which has the highest hitting rate up to 69%.

1. Introduction

Although price fluctuation is assumed almost to be a perfect random walk [1] in most approach in financial engineering [2], high frequency behavior of prices for less than a minute is known to largely deviate from the perfect randomness [3,4]. This fact motivates us to predict the immediate future of the financial time series based on its history. However, the price motions are highly random and complex, which makes the prediction very hard.

We report in this paper some results of our attempt to develop investment strategy based on an evolutional technique inspired by Lindgren's model [5] originally presented in the context of prisoner's dilemma that generates longer strategies by means of duplications and errors.

2. Model

We assume that each agent has its own history (H) and strategy (S), and use them to determine its next motion.

A history H is represented by a string of {0,1,2}, that records the motion of the last N ticks such as

$$H = h_0 h_1 h_2 h_3 \cdots h_{(N-1)}$$

$$h_i \in \{0,1,2\}$$

Here h_i represents the move at i-ticks before the current move and 0 stands for a 'down' motion relative to the previous motion, 2 an 'up' motion relative to the previous motion, and 1 a steady position.

A strategy S is regarded as a genome, represented by a string of length 3^N, each site of which has a value of 0, 1, or 2, representing 'Down', 'Flat', and 'Up' motion, respectively. This can be regarded as the leaves of a triplet tree of height N. Fig.1 shows a case of N=4. For example, a strategy such as 220*...* expects a 'Up' motion for the next tick when the history is either 0000 or 1000, and 'Down' motion when the history is 2000.

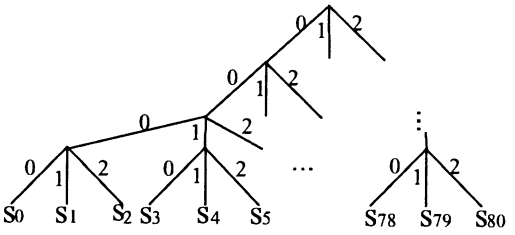


Fig.1 Strategy represented by a tree graph when the length of history is four.

Each agent (=strategy) performs a series of predictions based on its initial strategy. The next generation is determined by taking the rate of correct predictions to be the evaluation function representing the adaptability.

We first re-order the agents according to the descending order of adaptability (performance). Then leave the best 5 % agents to the next generation and discard the worst 5 %. The rest 90 % agents staying between the best 5 % and the worst 5 % receives point-mutation with a fixed rate before moving to the next generation. This process keeps the total number of agents unchanged. We have changed each element of a strategy s_i randomly to one of {0.1.2} with a fixed value.

3. Simulation

We have used in this work tick data of foreign rate offered by CQG from January 2, 1995 to April 30, 2001. The first 1,000,000 points

consisting of

(1) 457,876 down motions
(2) 63,202 flat motions
(3) 478,922motions

We used 100 agents and chose the initial strategies randomly. Those agents evolve their strategies under changing environment given by the real data. After living one generation, i.e., after leaning the price fluctuation for one million ticks, all of them undergo a generation change. Generation change takes place according to the rule at the end of Section 2. We have chosen the rate of mutation to be 0.01 for each gene. A new set of agents repeat learning the same one million data until they reach the end of data and undergo another generation change. We have repeated this cycle for 2000 generations in order to educate the agents before proceeding to the next 100,000 points to test the learned strategy.

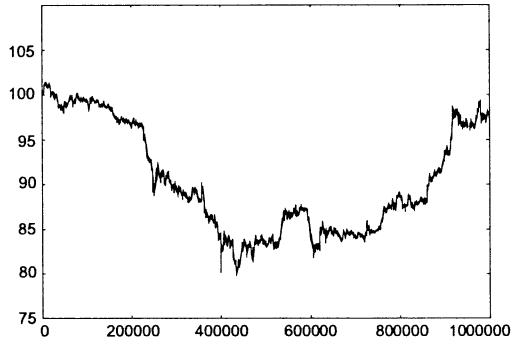


Fig.2 Price fluctuation of foreign exchange rates (tick data of US Dollar vs. Japanese Yen from January 2, 1995 to April 30, 2001)

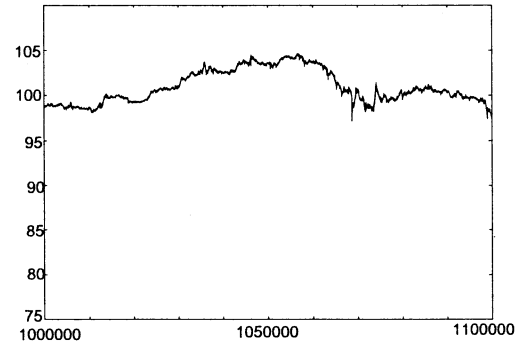


Fig.3 The 100,000 points following the data shown in Fig.2, which is used for testing the learned strategy by using the data in Fig.2.

4 Results

The trained result is shown in Table 1 and Fig. 4, where we listed the performance (in %) of the best strategy created after 2000 generations. The history lengths is chosen from 1 to 6. We can see that the strategy of longer history have higher rate of adaptability.

Table 1 Rate of correct prediction by the best strategy after 2000 generation of learning for various length of history (for the training data)

History length	Hitting rate(%)
1	68.492
2	68.807
3	68.851
4	68.871
5	68.875
6	68.886

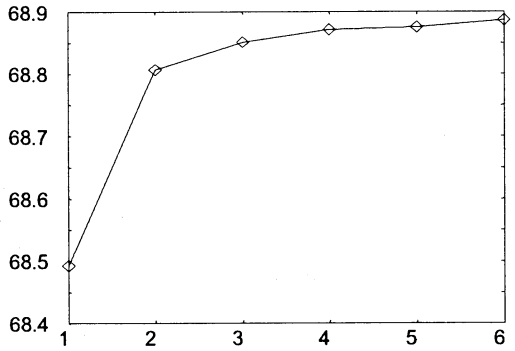


Fig.4 Rate of correct prediction (in %) of the best strategy after 2000 generations is plotted as a function of history length

Table 2 and Fig.5 show the result of testing the best strategy applied to new data of length 100,000 points, following the training data. We can see that the hitting rate increases as the history length grows up the four. However, the performance did not improve greatly if we increase the history length longer than four.

Since the best history length seems to be around 5, we will focus on this particular case in the following discussion. Note that the length of strategy in this case becomes 3^5=243.

Table 2 Rate of correct prediction by the best strategy after 2000 generation of learning for various length of history applied to the test data following the training data.

History length	Hitting rate (%)
1	68.655
2	68.811
3	68.824
4	68.860
5	68.865
6	68.847

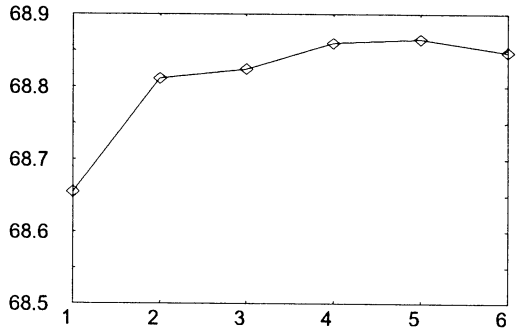


Fig. 5 Rate of correct prediction (in %) of the best strategy, when it is applied the test data following the training data, is plotted as a function of history length.

Figure 6 shows the rate of correct predictions made by the best strategy, 20-th best strategy, 50-th best strategy, and 100-th best strategy out of total 243 strategies as a time series of up to 2000 generations.

Before training, the rate of correct prediction of best strategy is 51%, the worst is 22%, and 37% on the average, which is almost equivalent to one third which is expected to be the performance of the random strategy. Each strategy quickly evolves its adaptability to the environment during the first 500 generations and continues evolving until 1000 generations. However, the evolution seems to reach a saturation point after 1200 generations and the rate of correct prediction does not grow further and the performance did not change much until we stopped our simulation at 2000 generations.

At that point, the best performance was 68.875 %, which is shown in Table 3. Table 4 shows the performance of the best genome (strategy) when applied to the test data.

We can see that the performance of the best strategy applied to the test data is the same level as that of the training data.

Table 3 Performance of the best strategy

688,754		Foreign Exchange Rate		
		Down	Flat	Up
Prediction	Down	346,387	33,006	115,732
	Flat	209	348	124
	Up	132,326	29,849	342,019

Table 4 Performance of 100,000 days

68865		Foreign Exchange Rate		
		Down	Flat	Up
Prediction	Down	34,623	3,239	11,915
	Flat	5	7	5
	Up	13,366	2,605	34,235

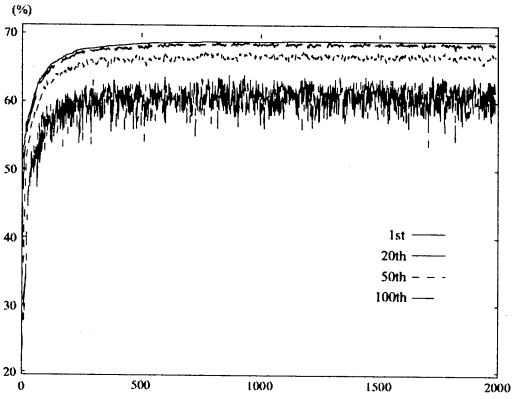


Fig. 6 Rate of correct predictions made by the best strategy (solid line), 20-th best strategy (dashed line), 50-th best strategy (dotted line), and 100-th best strategy (sparsely dashed line) for 2000 generations.

5. Conclusion

In this paper we attempted to obtain better investment strategy starting from the randomly chosen initial values by training them under changing environment of the market offered by the real financial data in an evolutional method.

We have obtained correct prediction for the direction of tick-wise motion of foreign currency exchange rate with a high rate of approximately 70% that is twice as large as the random prediction.

Note that the strategies in our model strictly depend only on its history of past few ticks. This reminds us of our previous work on the memory of tick data extending up to 3-4 in various aspects including conditional probabilities, correlation lengths, and mutual information. [3,4]

In this paper we have confirmed the fact that the performance grows as the length of history up to five. Whether the performance saturates somewhere or it grows monotonically is an open problem to be studies in future. Also we expect to improve the performance if we incorporate other factors than the price motions.

Acknowledgments

This work is supported in part by Scientific Grant-in -Aid by the Ministry of Education, Culture, Sports, Science and Technology of Japan (C2:14580385)

References

[1] Bachelier L. "Théorie de la speculation", Doctor Thesis. Annales Scientifiques de l'Ecole Normale Sperieure III-17:21-86; Translation

(1964) : P.H. Cootner(Ed.) the Random character of stock market prices, MIT Press (1900) 17-18
[2] e.g. Thomas Mikosch, "Elemenary stochastic calculus with finance in view", (World Scientific, 1998)
[3] Mieko Tanaka-Yamawaki and Shinya Komaki, "Characteristic Features of High Frequency Financial Time Series", Proc. of The Eighth Int. Symp. on Artificial Life and Robotics (AROB8th '03), Beppu, Oita, Japan, 24-26 January, 2003, pp.74—77
[4] Mieko Tanaka-Yamawaki, "Stability of Markovian Structure Observed in High Frequency Foreign Exchange Data", Ann. Inst. Statist. Math, Vol.55, 2003, pp.437-446.
[5] Kristian Lindgren,"Evolutionary Phenomena in Simple Dynamics" in ARTIFICIAL LIFE II (Christopher G.Langton, Charles Taylor, J.Doyne Farmer, Steen Rasmussen, ed.) (1990) Addison Wesley.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A Sub-goals Adaptation Method for Quick Reinforcement Learning in Varied Environments

Kenji Kondo, Koichiro Yamauchi, Takashi Omori

Division of Systems and Information Engineering, Graduate School of Engineering, Hokkaido University
Kita13Jyou Nishi8Choume, Kita, Sapporo, 060-8628, Japan
{kenji, yamauchi, omori}@complex.eng.hokudai.ac.jp

Keywords: hierarchical reinforcement learning, sub-goal, multiple tasks

Abstract

Although reinforcement learning system is very effective learning scheme in ambiguous environments, many of conventional reinforcement learning methods is for solving a single task. Therefore, they are hard to adjust quickly to the environmental changes. To adjust the varying environments quickly, the systems should re-use their modules acquired in the past experiences to accelerate learning. Our system employees a certain module to solve the corresponding sub-goal and finds the route between the sub-goal and the final goal. The system optimizes sub-goal modules sequentially by using an objective function which measures each sub-goal module's ratio of contribution to the tasks. The system makes each sub-goal slightly move to maximize the objective function every certain time lengths. In the experiments, we show that the system find appropriate sub-goals which are shared by several different navigation tasks in grid world. Moreover, we also show that the sub-goals are adaptively modified to fit the dynamically changing tasks.

1 Introduction

Although reinforcement learning system is very effective learning scheme in ambiguous environments [1], almost all of conventional reinforcement learning methods are for solving a single task. Therefore, they are hard to adjust quickly to the environmental changes. To adjust the varying environments quickly, the systems should reuse their modules acquired in the past experiences to accelerate learning.

Many researchers have suggested hierarchical reinforcement learning systems [2, 3, 4, 5, 6, 7]. These systems hierarchically construct modules to solve sub-tasks into which a task is divided. In most of those systems, each of modules has the ability to solve the corresponding sub-goal. To solve a new task, those systems find the route between a certain sub-goal as a module and the final goal of the task. Therefore, if the route is easy to find, the learning for the task is much faster than the learning without reusing the module. Note that each module, due to the nature of hierarchy, also can employ others to solve its own sub-goal and the systems thereby have high reusability. The question now arises: how sub-goals of those modules are set? In Most of those systems, these are set by system designers. It is necessary for those systems to automatically determine the ability of modules in the unknown or dynamic situations.

The Nested-Q-Learning (NQL) system [4] can set sub-goals automatically. The sub-goal generation process is realized after preliminary learning. The sub-goal generation method, however, does not include the modification of sub-goals so that the system cannot adapt to the change in distribution of final goals.

To solve the problem, we improved the original NQL method to optimize sub-goal modules sequentially. To optimize each sub-goal, the system uses an objective function which measures each sub-goal module's ratio of contribution to the tasks.

In the experiments, we show that the system finds appropriate sub-goal module which is shared by several different navigation tasks in grid world. The system sequentially learns the several navigation tasks. As a result, the sub-goal was moved to a certain position and became to be shared by several navigation tasks. Moreover, we also show that the sub-goal is adaptively modified to fit the dynamically changing tasks.

2 Hierarchical Reinforcement Learning

In conventional Q-learning systems, an agent learns a task using a monolithic Q-function. Therefore, these systems are not suitable to learn multiple tasks. If there is the relativity between the tasks, systems efficiently learn them by using common knowledge between them. Hierarchical reinforcement learning is capable of doing so. Hierarchical reinforcement learning systems, which have been studied by many researchers, define the modules as the abstract actions, in addition to the primitive actions.

We employ the NQL system, which is one of hierarchical reinforcement learning systems, for constructing hierarchical modules. In the system, modules have the ability to solve the corresponding sub-goal. The characteristic of the system is the nature that one module can

employ the other modules. One module exclusively controls an agent at a certain situation. If one module invokes another module, the former module is suspended until the invoked module solves its own sub-goal. If the sub-goal is solved, the suspended module is resumed to solve the remained path between the sub-goal and the resumed module's sub-goal.

Mathematically, the cooperation of the modules is described as follows. Let S be a set of states, A be a set of primitive actions and C be a set of modules. The i[th] module stochastically controls an agent based on the Q-function $Q_i(s,u)$, where $s \in S$ and $u \in \{A, C\}$. Upon performing the selected action, u, be it a primitive action or a module, the agent advances from state s to the next state s' and incurs a reinforcement signal, r . This reinforcement signal is defined as follows,

$$r = \begin{cases} -R & \text{if u is a primitive action} \\ \sum_{k=0}^{K} r^u(k) & \text{if u is a module} \end{cases}$$

where $\sum_{k=0}^{K} r^u(k)$ denotes the reinforcement signal from the invoked module summed over the number of steps, K, required to perform the module, u, and R is a positive constant that reflects the cost of performing the primitive action. To construct the Q function, the error, e, is defined to be,

$$e = r + \gamma \cdot \max_u Q^i(s',u) - Q^i(s,u)$$

where γ is the temporal discount factor $0 < \gamma \leq 1$ and $\max_u Q(s',u)$ is the current prediction of the maximum total future reinforcement remaining by arrival to the sub-goal when the agent leaves state s'. This error is used to adapt the evaluation function,

$$Q^i(s,u)_{new} = Q^i(s,u)_{old} + \alpha \cdot e$$

where α is the learning rate $0 < \alpha \leq 1$.

The original NQL system has the bottom-up function which is used in order to call the highest module. But we don't use the bottom-up function, instead, we define the extra modules each of which has the final goal of the given task as the sub-goal. When a task is given, an agent invokes the extra module corresponding to the task at the first, then the extra module employs the modules, which are not the extra modules, to solve the own sub-goal. Note that completing the extra modules means completing the task. Our sub-goals modification method is not applied to these modules.

3 Modification of Sub-goals

The learning performance for a new task depends on the functions of the modules which are reused. In most hierarchical reinforcement learning systems, the functions of modules are hand designed. On the other hand, in the original NQL system, the functions of the modules are automatically defined by setting these sub-goals. They will be selected based on two criteria. First, high changes in reinforcement gradient and secondly, a high frequency of visits to a particular state.

The sub-goal generation process is realized after preliminary learning. The sub-goal generation method, however, does not include the modification of sub-goals so that the modules cannot adapt to the change in distribution of tasks.

To solve the problem, we propose a novel sub-goal finding method which modifies modules sequentially. To optimize each sub-goal, we use an objective function which measures each module's contribution ratio to tasks.

3.1 New Sub-goal Modification Algorithm

The new system uses an objective function which measures contribution ratio of each sub-goal. Our system acquires the high available modules by moving their sub-goals based on the contribution ratio. The objective function has two terms: the first term denotes the running time of a module and the second term denotes extra cost of a module for the tasks. The i[th] module's ratio of contribution in the j[th] episode, $O_i(j)$, is,

$$O_i(j) = a_i - l \cdot b_i$$

where l is the constant, a_i and b_i are,

$$a_i = K_i / K$$

$$b_i = res_i \cdot \left| TotalR_{opt} - TotalR_{cur} \right|$$

where K_i is the number of steps required to perform the module and K is the number of steps required to solve the current task. And $TotalR_{cur}$ is the total cumulated cost to solve the current task, $TotalR_{opt}$ is the $TotalR_{cur}$ which is the least cost which the agent has experimented by the episode and res_i is responsibility of the module for this extra cost. res_i is defined to be,

$$res_i = \begin{cases} 1 & \text{if the module is the highest to the task} \\ 0 & \text{otherwise} \end{cases}$$

The objective functions of all modules are calculated every episode. And also each criterion for the movement of their sub-goals, O_i, are updated as follows,

$$O_i^{new} = O_i^{old} + \beta \cdot (\Delta O_i)$$

where β is the learning rate and ΔO_i is represented as

$$\Delta O_i = O_i(j+1) - O_i(j)$$

The direction of movement for a sub-goal is determined based on ΔO_i, which is calculated every a certain number of episodes (M episodes). The i[th] module's movement vector, $\mathbf{d}_i$, is defined to be,

$$\mathbf{d}_i = \begin{cases} \mathbf{d}_i^{old} + \mathbf{rand} & \text{if } \Delta O_i > \varepsilon \\ \mathbf{rand} & \text{if } |\Delta O_i| \le \varepsilon \\ -\mathbf{d}_i^{old} + \mathbf{rand} & \text{if } \Delta Oi < -\varepsilon \end{cases}$$

where $\mathbf{d}_i^{old}$ is the previous movement vector, **rand** is the vector whose elements are small values and ε is the small positive constant. The initial states of sub-goals are set random positions over the state space.

3.2 Techniques for Quick Adaptation of Modules

As mentioned before, each sub-goal is modified according to the contribution ratio. If a sub-goal is modified, the corresponding module also modifies its Q-table for the new modified sub-goal. In other words, the module is modified to adjust to the new sub-goal. However, this raises a serious problem. If another module employs the module of the modified sub-goal, it does not work well until the module employed completes the modification of its Q-table to solve the new modified sub-goal well. Hence, the system can not calculate the contribution ratio of each module precisely until all modules completes the modification of its Q-tables. Therefore, it is important for the modules of the system to adapt quickly the new sub-goals when module's sub-goal is moved.

To overcome the problem, our system uses techniques for the quick adaptation of modules when their sub-goals moved. These techniques need a condition that the amount of a sub-goal's movement is small. The system makes a certain small area around the new sub-goal, which is called All Goal Updating (AGU) [5] area around the modified sub-goal, where states are within some numbers of steps from the modules' sub-goal. From any points in the AGU area, the module can control the agent to reach a new modified sub-goal.

The AGU is a method for updating multiple Q-tables for multiple goals simultaneously. In the AGU, whenever an agent performs a primitive action, the system updates the Q-tables for all defined goals by using a state transition sample, (s, a, s'), that the action causes.

In this paper, we represent the Q-tables which are made for i[th] module as AGM_i and a set of states in the module's AGU area as S_i^{agm}. And the Q-table corresponding with a goal g for AGM_i is represented as AGM_i^g. Note that their Q-tables handle only primitive actions.

Note that the i[th] module is not learned well for the new sub-goal, $newsg_i$, for a while after its sub-goal was moved. Our system solves the new sub-goal by using AGM_i for the while. When the i[th] module controls an agent, it acts based on Q^i outside S_i^{agm} while it acts based on the $AGM_i^{sgnew_i}$ inside S_i^{agm}. In this way, the module can solve the new sub-goal easily before the module is not learned well for the new sub-goal. The Q^i-table is updated while an agent applies AGM_i for the while (N episodes). Then, an agent stops applying AGM_i.

S_i^{agm} is updated by using the buffer that an agent stores the last some histories of visited states. When an agent achieves the i[th] module's sub-goal at the time, t, $S^{buffer}(t)$ is added to S_i^{agm}, where $S^{buffer}(t)$ is a set of states stored in the buffer at t. And states which are not added to for the certain number of episodes are removed from S_i^{agm}.

Thus, the modules learn its AGM in preparation for the movement of their sub-goal for the quick adaptation. Therefore, the system can modify the modules without the particular degradation of performance solving their sub-goal. These techniques contribute to make the system quickly acquire available modules for tasks.

4 Experimental Results

In the experiments, we show that the system find appropriate modules which are shared by several different navigation tasks in grid world. We also show that the modules are adaptively modified to fit the dynamically changing tasks.

In this simulation, we used the two dimensional environment of Figure 1.

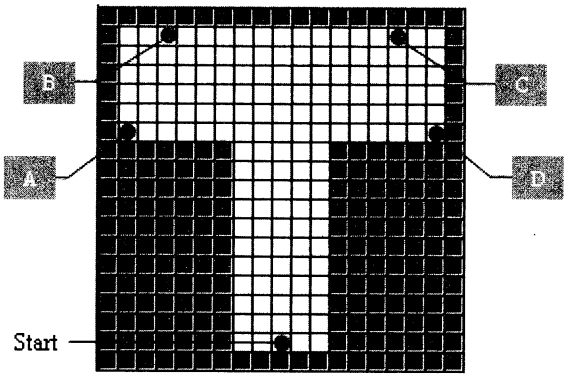


Figure 1

The learning agent has primitive actions to move up, down, right or left, and it was placed in this grid world. There are four types of learning tasks; each of them has the final goal and the stating location indicated location in Figure 1. They were labeled A, B, C and D. The system passes through the two periods. In the first period, the agent solves the task, which is selected from among the four tasks randomly, at an episode. In the second period, the agent solves the task, which is selected from either A or B randomly, at an episode. In this paper, we show the case where there is a single module to be modified. We used the following parameter settings: α = 0.2, β = 0.01, γ = 1, ε = 0.01, R = 1, M = 200, N = 100 and the buffer size to store states for the AGM is 5. The initial state of the sub-goal was randomly set at the location indicated in Figure 2(a).

Figure 2(a) and Figure 2(b) illustrate the situation

where module's sub-goal moves. The location which is darkly represented means the large number of times of setting the sub-goal there. Figure 2(a) is in the first period. This shows that the sub-goal moved around the location where the agent can pass at the four tasks in common. In other words, the module is able to be shared by them. And Figure 2(b) is in the second period. This shows that the system acquired the more available module for the two tasks, while the module acquired in the previous period was also available for them. Thus, this system is capable of adapting the module to the change in distribution of tasks.

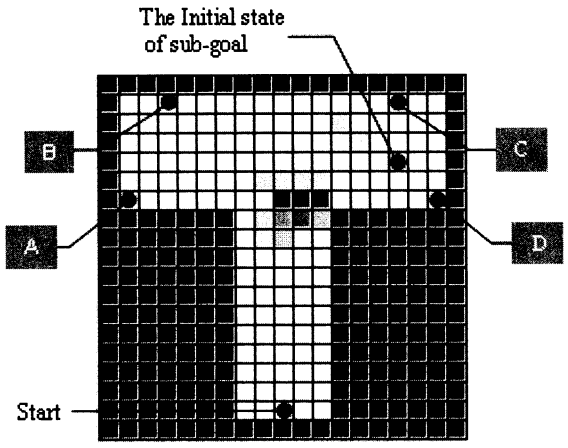


Figure 2(a)

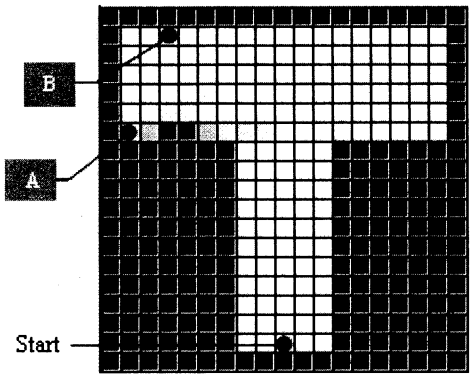


Figure 2(b)

5 Conclusions

We propose a novel sub-goal finding method which modifies modules sequentially. In the experiments, we showed that our system finds appropriate module which is shared by several different navigation tasks in grid world.

In this paper, the numbers of modules, which are shared by the tasks, were restricted to one. But the system can employ multiple modules. In that case, the system will have higher availability, however it will need higher learning cost. We will try various tests to make the system acquire hierarchical modules in the future work. We will also consider the convergence of modifying modules.

References

[1] Sutton, R. S and Barto, A. G. (1998) " Reinforcement Learning: An Introduction", MITpress.

[2] Dietterich, T. G. (2000) "Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition", Journal of Artificial Intelligence Research 13, pp.227-303

[3] Digney, B (1996) "Emergent hierarchical control structure: Learning reactive/hierarchical relationships in reinforcement environments", In From Animals to Animals 4: The Fourth Conference on Simulation of Adaptive Behavior, MIT Press, 1996.

[4] Digney, B (1998) "Learning hierarchical control structure from multiple tasks and changing environments", In From Animals to Animals 5: The Fifth Conference on Simulation of Adaptive Behavior, MIT Press

[5] Kaelbling, L. P. (1993) "Hierarchical reinforcement learning: Preliminary results", In Proceedings of the Tenth International Conference on Machine Learning, pp.167-173

[6] Singh, S. P. (1992) "Transfer of learning by composing solutions of elemental sequential tasks", Machine Learning 8, pp.323-339

[7] Sutton, R. S., Precup, D. and S. Singh (1999) "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning", Artificial Intelligence 112, pp.181-211

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Approaching motion control of the Humanoid Robot by using Linear Visual Servoing

Kazuya Okamoto, Kengo Yamaguchi, Noriaki Maru

Wakayama University
Faculty of Systems Engineering
Sakaedani 930 Wakayama, JAPAN

Abstract

This paper presents a method for approaching motion control of the Humanoid Robot by using linear visual servoing (LVS). Approaching motion control is realized by the approaching motion of the body to the target object keep a desire distance using virtual maker. LVS is based on linear approximation of the transformation between binocular visual space and joint space. It is very robust to calibration error, especially to camera angle error, because it uses neither camera angles nor joint angles to calculate feedback command. Some experimental results are presented to show the effectiveness of the proposed method.

1. Introduction

Vision is indispensable for the Humanoid Robots that coexists with human being and perform specified tasks in a changing environment. Various kinds of mechanism of visual feedback have been proposed and they are called visual servoing [1]. Especially, visual servoing is proposed as a robust control method for calibration error. But conventional visual servoing schemes are not so robust to calibration error, because they use both image Jacobian matrix which includes camera angles and robot Jacobian matrix which includes joint angles to calculate feedback command.

Mitsuda et. al proposed a simple visual servoing scheme called linear visual servoing (LVS) for the Humanoid robots [2][3]. LVS is based on linear approximation between the binocular visual space and joint space. The method is very robust to calibration error, especially to camera angle error, because it uses neither camera angles nor joint angles to calculate feedback command. Hence, it is especially suitable for the Humanoid Robots which use active stereo vision. That is, it is possible to turn cameras to facilitate visual processing, even if the manipulator is under control by visual servoing. Furthermore, the amount of the calculation is very small compared to the conventional visual servoing schemes, because it only needs both the time-invariant constant matrix and the image coordinates of the feature points.

In this paper, we propose a method for approaching motion control that a constant distance was keeping of the Humanoid Robot by using LVS. Approaching motion control is realized by the approaching motion of the body to the target object keep a desire distance. Approaching motion control by using LVS is based on linear approximation of the transformation between binocular visual space and motion space of the Humanoid Robot. Motion space is defined by translational velocity and rotational velocity of the robot coordinate system which is attached at the center of gravity of the Humanoid Robot. Approaching motion control that a constant distance was keeping by LVS is based on linear approximation of the transformation between binocular visual space and the target of imagination of the humanoid robot. We will show some experimental results which demonstrate the effectiveness of the proposed method.

2. Approaching Motion Control by using LVS

2.1 Model of the Humanoid Robot

Fig.1 shows the Humanoid Robot. The Humanoid Robot consists of two independently driven wheels and two casters. The two cameras pan and tilt independently. We show the parameters of the Humanoid Robot in **Table.1**.

The binocular visual space is defined as the vergence angle γ and the viewing direction θ , δ (see **Fig.2**). The binocular visual coordinates $\mathbf{V} = (\gamma, \theta, \delta)^T$ of a feature point projected on the camera image planes are

described as

$$\mathbf{V}=\begin{pmatrix} \alpha_L-\alpha_R \\ (\alpha_L+\alpha_R)/2 \\ \alpha_D \end{pmatrix}+\begin{pmatrix} (X^L-X^R)/f \\ (X^L+X^R)/2f \\ (Y^L+Y^R)/2f \end{pmatrix} \quad (1)$$

where α_L, α_R and α_D are the camera angles and (X^L,Y^L) and (X^R,Y^R) are the coordinates of a feature point in the left and right image respectively.

Table.1 Parameters of humanoid robot

Parameter	Value (mm)
Hight Length (H)	1300
Width Length (W)	770
Depth Length (D)	875
Baseline Length / 2 (E)	60
Camera Offset (N)	167
Maker Point (M)	300

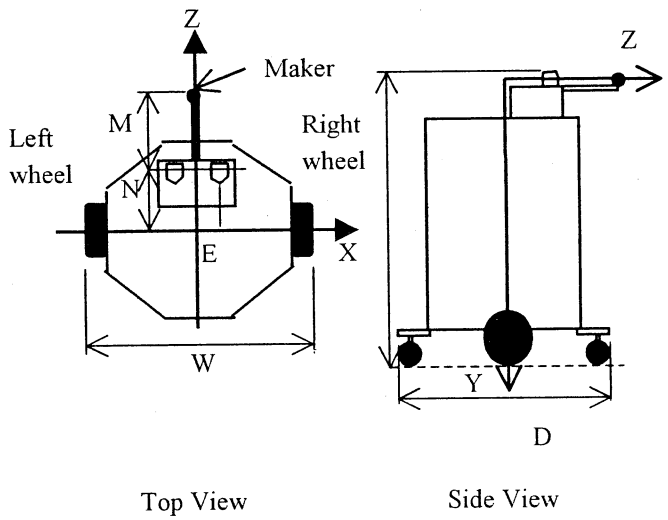


Fig.1 Model of the Humanoid Robot

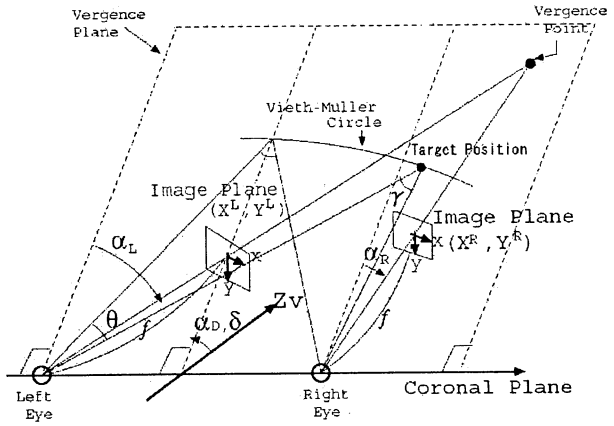


Fig.2 Binocular visual space

2.3 Linear Approximation of the Inverse Kinematics of Motion

Fig.3 show the motion space of the humanoid robot projected onto Cartesian space. We linearize these transformation using the least squares approximation within a region defined as 300[mm] ≦ R ≦ 3000[mm], -30[deg]≦ ϕ ≦ 30[deg].

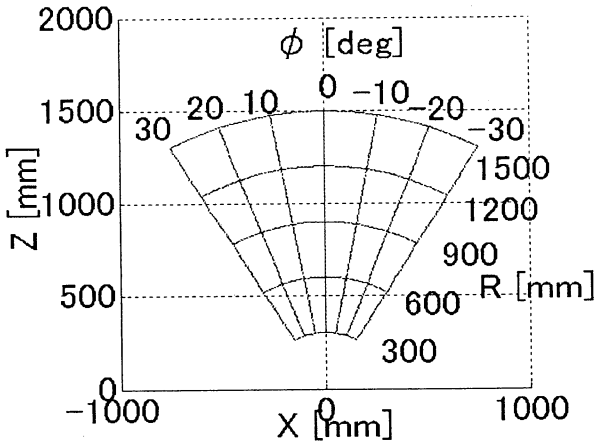


Fig.3 Motion space projected onto Cartesian space

Then the transformation from the binocular visual space to motion space are given by

$$\mathbf{j_m} = \mathbf{R_m}\,\mathbf{V} + \mathbf{C_m} \quad (2)$$

where $\mathbf{V} = (\gamma, \theta, \delta)^T$ and $\mathbf{j_m} = (R, \phi, 0)^T$, $\mathbf{R_m}$: matrix (3×3), $\mathbf{C_m}$: vector(3×1), r: translational velocity, ϕ :rotational velocity.

The least-squares approximation using binocular visual space result in

$$\mathbf{R}_m = \begin{pmatrix} 489.204 & 0.000 & 0.000 \\ 0.000 & 1.130 & 0.000 \\ 0.000 & 0.000 & 0.000 \end{pmatrix} \quad (3)$$

$$\mathbf{C}_m = \begin{pmatrix} 301.75 \\ 0.000 \\ 0.000 \end{pmatrix} \quad (4)$$

2.4 Approaching Motion Control by using LVS

Linear visual servoing for approaching motion control is given by

$$\mathbf{u}_m = -\lambda_1 \mathbf{R}_m (\mathbf{V}-\mathbf{V}_d),$$

$$= -\lambda_1 \mathbf{R}_m \begin{pmatrix} \{(X^L-X^R)-(X^L{}_d-X^R{}_d)\}/f \\ \{(X^L+X^R)-(X^L{}_d+X^R{}_d)\}/2f \\ \{(Y^L+Y^R)-(Y^L{}_d+Y^R{}_d)\}/2f \end{pmatrix} \quad (5)$$

where $\mathbf{u}_m$ is the control signals to mobile velocity controllers, $\mathbf{V}$ is the binocular visual coordinates of the maker at the humanoid robot, $\mathbf{V}_d$ is the binocular visual coordinates of a target marker and λ_1 is a scalar gain, $\mathbf{R}_m$ is the linear approximation matrix of the inverse kinematics obtained in the previous section.

2.5 Approaching Motion Control a desire by using LVS

Fig.4 shows the virtual maker of the Humanoid Robot. Linear visual servoing for approaching motion control that a certain distance was keeping is given by

$$\mathbf{u'}_m = -\lambda_1 \mathbf{R}_m (\mathbf{V}_v-\mathbf{V}_d),$$

$$= -\lambda_1 \mathbf{R}_m \begin{pmatrix} \{(X^L-X^R)-(X^L{}_d-X^R{}_d)\}/f - I \\ \{(X^L+X^R)-(X^L{}_d+X^R{}_d)\}/2f \\ \{(Y^L+Y^R)-(Y^L{}_d+Y^R{}_d)\}/2f \end{pmatrix} \quad (6)$$

where $\mathbf{u'}_m$ is the control signals to mobile velocity controllers, $\mathbf{V}_v$ is the binocular visual coordinates of the virtual maker of the humanoid robot, $\mathbf{V}_d$ is the binocular visual coordinates of a target maker and λ_1 is scalar gain, $\mathbf{R}_m$ is the linear approximation matrix of the inverse kinematics obtained in the previous section, I is $\gamma_m - \gamma_v$.

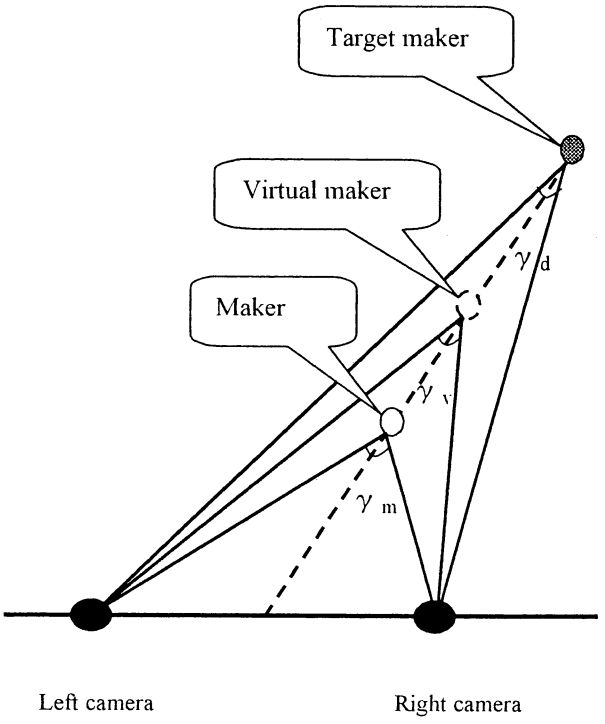


Fig.4 Virtual maker

3 Experiment

3.1 System Construction

Fig.5 shows the experimental system. We used two EVI-D100 (SONY Co. Ltd) as a stereo cameras which pan and tilt independently and IP5000 (HITACHI Co. Ltd) as an image processing and DOS/V computer. We attached a yellow marker in front of the stereo cameras as the maker of the Humanoid robot and used a red object to simplify image processing. We experimented the value of γ_v is changed into from 350(mm) to 1000(mm). The stereo images are converted in an image using a picture separator (Video Device Co. Ltd.) and input to IP5000. The stereo images are binarized by each color and the center of gravity is calculated to obtain the image coordi-

nates of both the maker of the Humanoid Robot and the target maker. Then the feedback commands are calculated and sent to the motor driver.

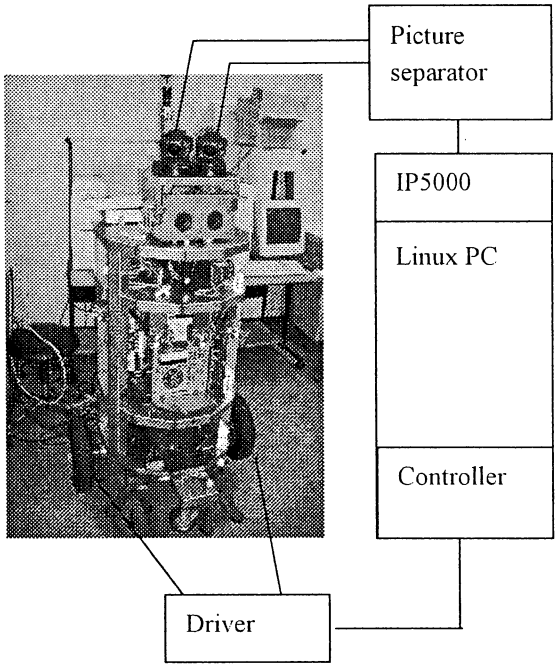


Fig.5 Experimental system

3.2 Experimental Results

Fig.6 show the distance between the maker and the target maker (300mm~1000mm) of the Humanoid Robot by using LVS. This figure indicates that LVS using binocular visual space is also effective to approaching motion control that a constant distance was keeping.

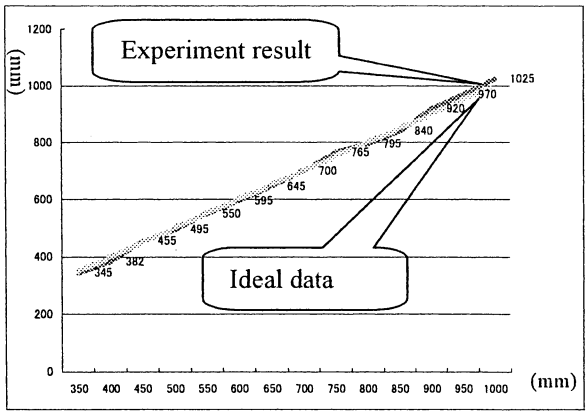


Fig.6 Experimental result of virtual maker

4 Conclusion

This paper presents a method for approaching motion control that a constant distance was keeping of the Humanoid Robot by using LVS. It is based on linear approximation of the transformation between binocular visual space and motion space. Approaching motion control by using LVS is based on linear approximation of the transformation between binocular visual space and motion space of the Humanoid Robot. Constant distance was keeping motion control by using LVS is based on linear approximation of the transformation between binocular visual space and motion space of the Humanoid Robot. We have shown some experimental results which demonstrate the effectiveness of the proposed method.

References

[1] L.E.Weiss, A.C.Sanderson and C.P.Neuman, "Dynamic sensor – based control of robots with visual feedback", IEEE J. Robotics Automation, vol.RA – 3,No.5, pp404-417, 1987

[2] T.Mituda, N.Maru, K.Fujikawa, F.Miyazaki" Visual Servoing based on Linear Approximation of the Inverser Kinematics" Journal of the Robotics Society of Japan vol.14 No.5 pp734 – 750 1996. (in Japanese)

[3] T.Mituda, N.Maru, K.Fujikawa, F.Miyazaki" Linear Approximation of the Inverser Kinematics by using a Binocular Visual Space" Journal of the Robotics Society of Japan vol.14 No.8 pp1145 – 151 1996. (in Japanese)

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Design of Vision-Based Motion Control System for Balloon Robot

Hisao Kadota* Hidenori Kawamura* Masahito Yamamoto*
Toshihiko Takaya** Ohuchi Azuma*

*Graduate School of Engineering, Hokkaido Univesity
Nishi 8, Kita 13, Kita-ku, Sapporo, Hokkaido, 060-8628, Japan
{kadota, kawamura, masahito, ohuchi}@complex.eng.hokudai.ac.jp

**RICOH SYSTEM KAIHATU COMPANY, LTD
Nishi 4, Kita 7, Kita-ku, Sapporo, Hokkaido, 060-0807, Japan
takaya@rsk-kitami.grp.ricoh.co.jp

Abstract

Balloon robots are attractive as new type indoor robots for various applications, because balloon robots can move safely and float long time in three-dimensional space with low energy. A Balloon robot control is very difficult by influences of strong inertia and slight air stream, but the adaptive control must be achieved for the realization of the practical balloon robots. In this paper we designed an indoor type balloon robot and aimed to go round several destinations as a motion control. The experimental results show that long flight of the balloon robot is achieved with PD-controller for going round several destinations.

keywords : balloon robot, PD-controller, motion control, camera

1 INTRODUCTION

Balloon robots have attracted attention as new indoor type robots in the aerial vehicles [1], [2]. Balloon robots can move safe and long time in three-dimensional space with low energy compared to the helicopters and airplanes because they float by buoyancy and don't have sharp rotors. Balloon robots that can recognize the environment are indispensable for various applications e.g., observation where human intervention is dangerous or expensive, guidance in the buildings and amusement flights.

Indoor type balloon robots have various advantages, but they have several disadvantages too. A balloon robot can not carry enough output devices and sensors because a balloon robot floats by filling the helium gas in the body. Furthermore a balloon robot control is difficult because a balloon robot is subjected to strong inertia and influenced slight air stream. We try to design a balloon robot system and achieve the adaptive control for the realization of a practical balloon robot under above circumstances.

In this paper, we design a balloon robot in the view of indoor applications and aim to achieve a motion control of the balloon robot. Achievement of the motion control making the round several destinations is the key technology to realize autonomous balloon robot.

2 BALLOON ROBOT SYSTEM

2.1 Pillar-Type Balloon Robot

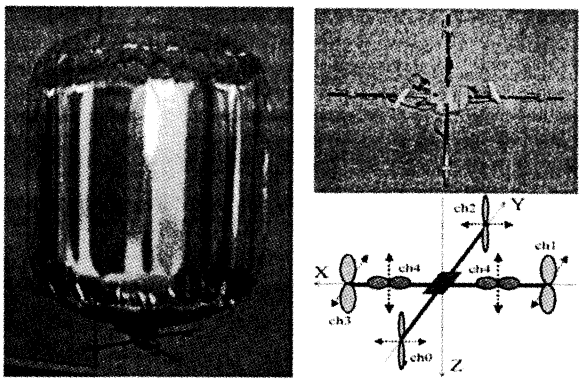


Figure 1: Balloon robot

The researches on a balloon robot have presented the airship-type balloon robot that has oblong body and does not have the propeller units to move just beside directly [1], [2]. But we design the holonomic pillar-type balloon robot as shown Figure 1 in the view of indoor applications, because the arbitrary direction movements and compact rotation are desired in narrow indoor space for a balloon robot. The diameter of the body is 88.0[*cm*] and the height is 86.0[*cm*] and the pillar-type body can have big payload because of large volume and realize the fair air resistance for all moving directions. The output device is attached on

the bottom of the body, and a camera is attached facing downward on the under side of the output device. The balloon robot can freely move to desirable directions in three-dimensional coordinate system fixed to oneself, because every two of propellers are set for each axial direction to let the balloon robot control three-axes independently in coordinate system as shown in Figure 1. The balloon robot moves horizontal plane with four propellers set on the end of crisscross rod and each propeller can respectively generate a thrust for positive and negative directions. Two propellers, that are set on the central part of the crisscross rod, generate the thrusts for the vertical direction, and rotate positive or negative direction together.

The thrusts generated from each propeller can be adjusted in analog by changing the voltage given to propeller motors and the property of thrusts generated from each propeller are nonlinear. The proportional property of thrusts is ideal status for the complex actions but it is difficult to have such a property on the propeller motors. We attempt to generate proportional thrusts by adjusting the operating time of the propellers per sampling time $\Delta T[sec]$. In the approach, each propeller motor is given the voltages generating nearly same thrusts.

2.2 Control System

Autonomous flight of the balloon robot is achieved by the control system as shown in Figure 2. The control system is simply divided into three parts, body part, control part and radio transmission part. The control part is placed outside because the balloon robot is not large enough for carrying the computer. In the control part, the output voltages given to the each propeller motor are decided by processing the sensory information transmitted from the body part. The output voltages are transmitted to the radio transmission part through D/A converter and then the control signals are transmitted to the body part.

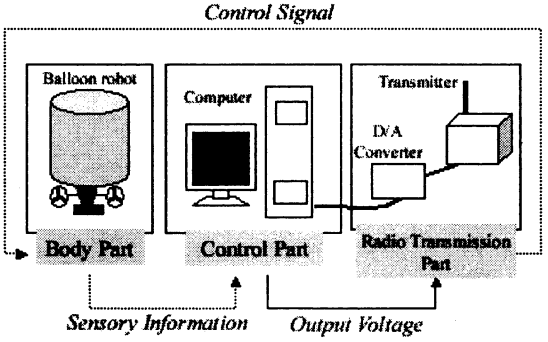


Figure 2: Outline of the Control System

3 MOTION CONTROL

The practical balloon robot control is active topic of our research and we have achieved the fundamental position control, moving to the target position and station keeping [3]. As a next step we aim at the achievement of going round the several destinations as the motion control.

3.1 Experimental Environment

In the application fields of the balloon robot, motion control is important technology, e.g., for the going round several destinations, path planning in the complicated environment. In this paper we aim at the achievement of going round the several destinations as the motion control. We experiment in the environment as shown Figure 3. In the experiment the balloon robot must get the positional information of the destinations to go round them. We use the vision camera facing downward to let the balloon robot perceive the positional information of the destinations and we see three red circles as the destinations. Three red circles are placed on the floor with same interval 150$[cm]$ and the destinations are set over each circle. Each diameter of the circles are 70, 50, 30$[cm]$. The balloon robot gets the positional information by recognizing them.

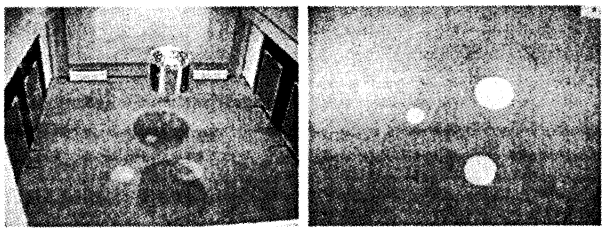


Figure 3: Experimental Environment

3.2 Recognition of the Environment

We use the red circles as a landmark to get three-dimensional positional information of the destinations. In the camera coordinate system where the origin is the image center, the circle center (xc, yc) changes with relative position from the balloon robot to the circle. The (xc, yc) is used to control the horizontal motion. The balloon robot can use the diameter $d[pixel]$ to control vertical motion, because it changes with the height of the balloon robot position from floor. Therefore the balloon robot must calculate the (xc, yc) and d to get the positional information of the destinations. We describe shortly the image processing algorithms, because it is not essential in this paper.

First, we extract the red regions in YUV space by thresholding method using the threshold values Y, U and V about the red circle given in preliminary exper-

iments. But, the extracted regions after thresholding method consist of other objects than the red circles.

Secondly, red circles regions are separated from the other regions by connected region labeling method [4] where the connected pixels are given the same labeling the raster scanning, and the pixels with same label are regarded as the same object region. After connected region labeling, we see the regions with same label that are less than a arbitrary threshold as the regions of noise. Then the remaining regions with same label are the different regions of the red circles.

Thirdly, as an internal point we select seed points surrounded by the $25[pixel]$ with same label for each circle, and then the image is scanned from the seed points to the every direction operating *Sobel filter*. The pixels whose values after operating the *Sobel filter* are more than the arbitrary threshold are regarded as the edge points.

Finally we calculate the relative deviation from the balloon robot to each destination. When n circles are detected in the camera, we can calculate the circle center (cx_i, cy_i) and the diameter d_i for each circle $i(1 \leq i \leq n)$ from detected three edge points in the camera coordinate system by using circular equations. Then the balloon robot can get the relative deviations, E_{Xi}, E_{Yi} and $E_{Zi}[cm]$ for destination over the circle i as follow.

$$E_{Xi} = cx_i \cdot l, \quad E_{Yi} = cy_i \cdot l, \quad E_{Zi} = h - set\_h_i$$

l is the approximate distance $[cm]$ per one pixel. l is calculated by using both the real interval $k_r = 150[cm]$ among the circles and average distance $k_a[pixel]$ of k_{ij} that is the distance between the circles center (cx_i, cy_i) and (cx_j, cy_j) in the camera coordinate system. h is the relation between the height of the balloon robot position from the floor and the average distance k_a, and the relation is given in the preliminary experiments. The set_h_i is the height of destination and is set per every circle.

$$l = \frac{k_r}{k_a}, \qquad h = \frac{31585}{k_a}$$

$$k_a = \frac{\sum k_{ij}}{{}_nC_2} \quad (0 < i < j \leq n)$$

Here, we can know the real diameter about circle i and distinguish among the circles with different diameter, because $l \cdot d_i$ is approximate to one of the real diameters of the circles. However, when the number of the circles in the camera is one or less, the balloon robot moves upward and extends the field of vision because l is not calculated.

When the current and next destinations given from circle i and j are detected in the camera, the balloon robot controls the angular to parallelize to the circle i to j path. The deviation $E_\theta[rad]$ is given as follow.

$$E_\theta = \arctan \frac{cx_j - cx_i}{cy_j - cy_i} \quad (-\pi < E_\theta \leq \pi)$$

3.3 Control Architecture

We implement PD-controller as the control architecture. The manipulated variable $m(t)$ at a certain time $t[sec]$ is given as the ratio of the motor operating time to sampling time ΔT as described in section 2. According as the relative deviation for distance E_d of each axial direction and angular E_θ detected from image processing, the manipulated variable for distance $m_d(t)$ and angular $m_\theta(t)$ are calculated as follow where K_P is the proportional gain and T_D is the derivative time.

$$m_d(t) = K_{P_d} \left\{ E_d(t) + T_{D_d} \frac{E_d(t) - E_d(t - \Delta T)}{\Delta T} \right\}$$

$$m_\theta(t) = K_{P_\theta} \left\{ E_\theta(t) + T_{D_\theta} \frac{E_\theta(t) - E_\theta(t - \Delta T)}{\Delta T} \right\}$$

We use the propellers $ch1$ and $ch3$ to control the angular, and $m(t)$ is given as follow.

$$m(t) = m_d(t) \pm m_\theta(t)$$

The other propellers is used to control only distance and $m(t)$ is given as follow.

$$m(t) = m_d(t)$$

Here, the propeller generates the thrust for positive direction at $m(t)$ being a positive value and for negative direction at negative value.

When p occasions are provided in ΔT for adjusting the thrusts by the motors operating time, $m(t)$ needs to be discrete value in multiples of $p/10$. The continued value $m(t)$ is transformed to discrete value $M(t)$ as follow.

$$M(t) = \begin{cases} [\,|m(t)| \cdot p\,] \,/\, p & \text{if} \quad |m(t)| < 1 \\ 1 & \text{otherwise} \end{cases}$$

In this control architecture, the propeller rotates from t to $t + M(t)\Delta T$. And then the propeller does not rotate until $t + \Delta T$ because the signal of motor operation converts from the positive or negative rotating voltage to stasis.

4 EXPERIMENT

4.1 Experimental Setup

We experiment for motion control of the balloon robot in the environment described in section 3. The

balloon robot is controlled to go round the three destinations where is fixed at the height of 400[cm] over the each circle center. The balloon robot repeats the follow tasks.

1. The balloon robot is controlled to keep within 50[cm] from the destination for 2[sec], and then the destination is switched to another point.

2. The balloon robot moves to the new destination.

Concretely speaking, the destination is repeatedly switched to the circle with the diameter of 70, 50 and 30[cm] in that order.

4.2 Result and Discussion

Figure 4 shows the relation between time lapse and the distance E from the balloon robot to the destination that is detected from camera and Figure 5 shows the transition of the destinations. The horizontal axis of the Figure 5 is time lapse and the vertical axis is the diameter of the circle that shows the landmark for the destination.

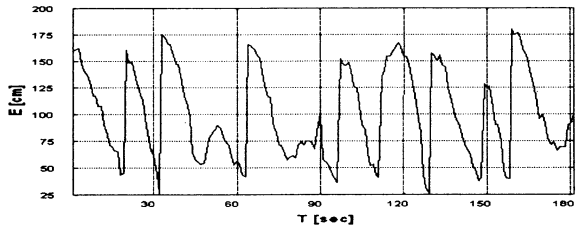


Figure 4: Distance from the Balloon Robot to Destination

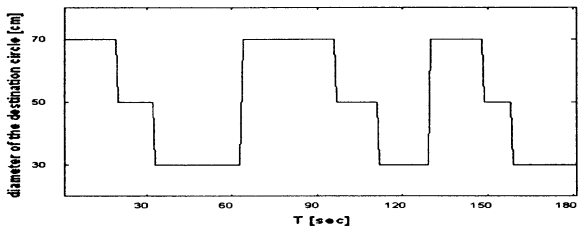


Figure 5: Relation between Time and Destination

As shown Figure 4 and 5, the distance from the balloon robot to the destination is long when the destination is just switched to the current destination. After that the distance from the balloon robot to the destination become gradually small, because the balloon robot is controlled to move to the destination. But the balloon robot sometimes passes over the destination greatly and then the balloon robot needs long time to keep near the destination because the balloon robot is subjected to great inertia. As the direct causes of passing over the destination, the realization of the neutral buoyancy to float motionless is difficult because the density of the helium and air changes with temperature variation. For this reason the Z-axial control is not stable.

But, we consider the balloon robot is controlled with accuracy because the balloon robot can go round 13 times and average 82.23[sec] among three destinations in 1080[sec] without losing the destinations.

5 CONCLUSION

We have presented the design of the pillar-type balloon robot system in the view of the indoor applications. The pillar type balloon robot has simplicity control and many advantages compared to airship-type balloon robot e.g., it can rotate compact and move to the desirable directions. For the practical applications of the balloon robot, we aimed at achievement of the motion control where the balloon robot goes round the three destinations. We use landmarks to perceive the destinations and the balloon robot get the positional information by recognizing them. As the control architecture, we implemented the PD-controller and achieved the fundamental motion control of the balloon robot.

Acknowledgment

Our special thanks are due to Mr.Yoshikatsu Henmi, Mr.Hitoshi Murakami and Mr.Yoshihiro Uchimi, Hitachi East Japan Solutions, LTD.; Dr.Dun Wu and Mr.Kenichi Akimoto, North Technology Co., LTD.; Mr.Mikitoshi Amano, TAKARA CO., LTD.; Dr.Keiji Suzuki, Hakodate Future University for considerable cooperation and valuable advice.

References

[1] J.C. Zufferey, D. Floreano, M. Leeuwen and T. Merenda. Evolving vision-based flying robots. In Proc. of the 2nd International Workshop on Biologically Motivated Computer Vision, pp. 592-600, 2002

[2] S. Zwaan, A. Bernardino and J. Santos. Vision based station keeping and docking for an aerial blimp. In proc. of IEEE/RSJ International Conference on Intelligent Robots and System, 2000.

[3] H. Kadota, H. Kawamura, M. Yamamoto, T. Takaya and A. Ohuchi. Vision-base Control of Autonomous Flying Airship Robot. In Proc. of ITC-CSCC'2003, pp. 628-631, 2003.

[4] J. Bruce, T. Balch and M. Veloso. Fast and inexpesive color image segmentation for interactive robots. In Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 3, pp. 2061-2066, 2000

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

A probabilistic approach to identify environmental models of mobile robots

Katsuyoshi Kanemoto[1], Junichiro Yoshimoto[2,1], and Shin Ishii[1,2]
[1]Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, JAPAN
[2]CREST, Japan Science and Technology Agency
{katsu-ka, juniti-y, ishii}@is.aist-nara.ac.jp

Abstract

To control mobile robots within the framework of statistical inference, precise models of state transition and observation are crucial. Although these models can be obtained if the robot observes its own trajectory using a roof camera for example, such a setting makes the robot's applicability narrow and the implementation costly.

In this study, we propose a method to identify the models from the robot's own observation. For simplicity, it is assumed that the robot takes only one action leading to the rotation of a specific and fixed angle and there is a cylinder as an observable object. In our method, we solve the identification problem by the maximum likelihood estimation with hidden variables. The estimators are obtained using the EM and particle filter algorithms.

Experimental results showed that our method was able to realize good estimation for synthetic and real environmental settings with little computation.

Keywords: Mobile robots, Khepera, Probabilistic model, System Identification, EM algorithm, Particle Filter.

1 Introduction

"Mapping" to obtain an environmental map and "Localization" to recognize the current location on the map are very important for mobile robots to solve every kind of tasks. In these years, many studies proposed to use statistical inference to solve these problems [1]. Such methods often need two models: (1) a state transition model representing how an action transforms the current state into the next one, and (2) an observation model representing what observation the current state provides. One way to obtain these models is to record the robot's trajectory using a roof camera for example, but such a setting makes the robot's applicability narrow and the implementation costly.

In this study, we propose a method to identify the state transition and observation models using the robot's own observation sequence and limited prior information on the environment. We deal with an environment where a small mobile robot 'Khepera' moves around a cylinder. The state transition process is formulated as a parameterized probabilistic model based on a geometrical property. The observation process is also formulated as a probabilistic model using an RBF network to allow a non-linear process. The EM [2] and particle filter [3] algorithms are used to identify the model parameters.

Experimental results showed that our method was able to realize good estimation for synthetic and real environmental settings with little computation.

2 The robot and the environment

Khepera is a small mobile robot with two wheels and eight infrared proximity sensors. Although the robot can go forward and/or rotate with two wheels, we assume that the robot takes only one action leading to the rotation of a specific and fixed angle for simplicity. Since the robot's center position never changes, the state is represented as a scalar value α_t indicating the robot's direction at time t (see Figure 1(a)). In addition, we assume that there is one cylinder as a landmark in the environment.

Figure 1 illustrates our problem setting. Figure 1(a) shows the global coordinate system to indicate the position and the direction of the robot. In this figure, x_c and y_c denote the cylinder's position along the horizontal and vertical axes, respectively. Figure 1(b) shows a local coordinate system to represent the position of each sensor on the basis of the center position and the frontal direction of the robot. x_{s_n} and y_{s_n} denote the position of the n-th ($n = 1, \ldots, 8$) sen-

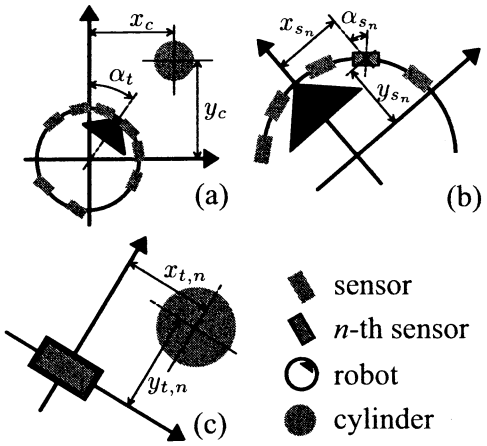


Figure 1: Geometrical relationship among the cylinder, the robot and its sensors.

sor along the horizontal and vertical axes in the local coordinate system. α_{s_n} denotes an angle between the vertical axis and the direction of the n-th sensor. Figure 1(c) shows another local coordinate system to represent the geometrical relationship between the n-th sensor and the cylinder on the basis of the position and the frontal direction of each sensor. $x_{t,n}$ and $y_{t,n}$ denote the position of the n-th cylinder at time t along the horizontal and vertical axes, respectively, in this local coordinate system. According to the geometrical transformation, we have the following relationship:

$$\boldsymbol{f}_n(\alpha_t) \equiv \begin{pmatrix} x_{t,n} \\ y_{t.n} \end{pmatrix} = \begin{pmatrix} \cos\alpha_{s_n} & -\sin\alpha_{s_n} \\ \sin\alpha_{s_n} & \cos\alpha_{s_n} \end{pmatrix} \times \left(\begin{pmatrix} \cos\alpha_t & -\sin\alpha_t \\ \sin\alpha_t & \cos\alpha_t \end{pmatrix} \begin{pmatrix} x_c \\ y_c \end{pmatrix} - \begin{pmatrix} x_{s_n} \\ y_{s_n} \end{pmatrix} \right).$$

We assume that x_c, y_c and $\{x_{s_n}, y_{s_n}, \alpha_{s_n} | n = 1, \ldots, 8\}$ are given as constants. Accordingly, $x_{t,n}$ and $y_{t,n}$ depend only on the current state α_t.

3 The process models

In this section, we formulate the environment as parameterized probabilistic models. The prior distribution of the initial state is assumed to be given by

$$P(\alpha_1) = \mathcal{N}_1(\alpha_1 | \mu_1, \sigma_1),$$

where μ_1 and σ_1 are the mean and variance of the initial state α_1, respectively. $\mathcal{N}_n(\cdot|\cdot,\cdot)$ denotes a n-dimensional normal distribution defined by

$$\mathcal{N}_n(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-n/2} |\boldsymbol{\Sigma}|^{-1/2} \times \exp\left[-1/2(\boldsymbol{x}-\boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})\right],$$

where $\boldsymbol{x}$ and $\boldsymbol{\mu}$ are n-dimensional vectors and $\boldsymbol{\Sigma}$ is an n-by-n covariance matrix. The prime symbol ($'$) denotes a transpose. According to the geometrical relationship, the state transition process can be represented as

$$P(\alpha_t | \alpha_{t-1}) = \mathcal{N}_1(\alpha_t | \alpha_{t-1} + \delta_\alpha, \sigma_\delta),$$

where δ_α is the amount of state change caused by a unit action. A white Gaussian noise with mean 0 and variance σ_δ is assumed to be added to the state transition process.

The observation process is modeled as follows. In an ideal environment with no noise, the observation of the n-th sensor at time t, $o_{t,n}$, is assumed to be given by a function $g(\cdot)$ of the relative position of the cylinder $\boldsymbol{f}_n(\alpha_t)$, i.e., $o_{t,n} = g(\boldsymbol{f}_n(\alpha_t))$. $g(\cdot)$ denotes the characteristics of the sensor. We assume that the function $g(\cdot)$ can be expressed by the following RBF network:

$$g(\boldsymbol{x}) = \sum_{i=1}^{N_n} v_i B_i(\boldsymbol{x}),$$

where N_n is the total number of units and B_i is a radial basis function defined by

$$B_i(\boldsymbol{x}) = \mathcal{N}_2(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \quad i = 1, \ldots, N_n .$$

$\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ are certain constants. If the parameters v_i $(i = 1, \ldots, N_n)$ are determined to specify the function $g(\cdot)$, the observation $o_{t,n}$ is given by

$$o_{t,n} = g(\boldsymbol{f}_n(\alpha_t)) = \sum_{i=1}^{N_n} v_i B_i(\boldsymbol{f}_n(\alpha_t)).$$

Note that this equation depends only on α_t after the parameter determination. By assuming that a white Gaussian noise with mean 0 and variance σ_o is added, the observation process is formulated as the following probabilistic model:

$$P(\boldsymbol{o}_t | \alpha_t) = \mathcal{N}_8(\boldsymbol{o}_t | \boldsymbol{B}(\alpha_t)\boldsymbol{v}, \sigma_o \boldsymbol{I}_8),$$

where $\boldsymbol{o}_t = (o_{t,1}, \ldots, o_{t,8})'$ and $\boldsymbol{v} = (v_1, \ldots, v_{N_n})'$. $\boldsymbol{I}_n$ is an n-by-n identity matrix and $\boldsymbol{B}(\alpha_t)$ is an $8 \times N_n$ matrix whose (i,j) element is $B_j(\boldsymbol{f}_i(\alpha_t))$.

According to the two models above, the joint distribution of state sequence $A_1^T = \{\alpha_1, \alpha_2, \ldots, \alpha_T\}$ and observation sequence $\boldsymbol{O}_1^T = \{\boldsymbol{o}_1, \boldsymbol{o}_2 \ldots, \boldsymbol{o}_T\}$, given model parameters $\boldsymbol{\theta} = \{\mu_1, \delta_\alpha, \boldsymbol{v}\}$, is given by

$$P(A_1^T, \boldsymbol{O}_1^T | \boldsymbol{\theta}) = \left[P(\alpha_1|\boldsymbol{\theta}) \prod_{t=2}^{T} P(\alpha_t | \alpha_{t-1}, \boldsymbol{\theta}) \right] \left[\prod_{t=1}^{T} P(\boldsymbol{o}_t | \alpha_t, \boldsymbol{\theta}) \right]. \quad (1)$$

In this study, the set of parameters $\{\sigma_1, \sigma_\delta, \sigma_o\}$ which indicates the noise level is assumed to be known.

4 EM algorithm

According to the maximum likelihood (ML) estimation method, the optimal model parameters $\boldsymbol{\theta}^*$ for given observations $\boldsymbol{O}_1^T$ is obtained as

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} P(\boldsymbol{O}_1^T|\boldsymbol{\theta}).$$

Since the probabilistic model (1) has hidden variables A_1^T, we calculate the ML estimator $\boldsymbol{\theta}^*$ using an iterative algorithm consisting of the following two steps.

- E-step
  Using the current estimator $\bar{\boldsymbol{\theta}}$, the posterior distribution of the state sequence A_1^T for the given observation sequence $\boldsymbol{O}_1^T$ is calculated. The result is given by $Q(A_1^T) = P(A_1^T|\boldsymbol{O}_1^T, \bar{\boldsymbol{\theta}})$.
- M-step
  The log-likelihood expected under $Q(A_1^T)$:

  $$L(\boldsymbol{\theta}) = \int Q(A_1^T) \log P(A_1^T, \boldsymbol{O}_1^T|\boldsymbol{\theta}) dA_1^T$$

  is maximized with respect to $\boldsymbol{\theta}$. The maximizer is then substituted for $\bar{\boldsymbol{\theta}}$, namely, $\bar{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} L(\boldsymbol{\theta})$.

This algorithm is called the EM algorithm. It is guaranteed that $\bar{\boldsymbol{\theta}}$ monotonously approaches the ML estimator $\boldsymbol{\theta}^*$ by repeating the E- and M-steps. The detailed implementations of the E- and M-steps are as follows.

4.1 E-step

Since the exact calculation of $P(A_1^T|\boldsymbol{O}_1^T, \bar{\boldsymbol{\theta}})$ is difficult due to the non-linearity of the assumed observation process, we approximate it using a particle filter algorithm shown by Figure 2. Here, α_t^i is a stochastic sample (particle) at time t and w_t^i is its weight. $x^i \sim p(x)$ means that a particle x^i is drawn from a probability distribution $p(x)$ independently. N_s is the number of particles at each time step.

By applying this algorithms, $Q(A_1^T)$ is estimated as an empirical distribution over the particles:

$$Q(A_1^T) = \frac{1}{N_s} \sum_{i=1}^{N_s} \left[\prod_{t=1}^{T} \delta(\alpha_t - \alpha_t^i) \right],$$

where $\delta(\cdot)$ is the Dirac's delta function. In the limit of $N_s \to \infty$, this approximation becomes exact.

Particle filter

- FOR($t = 1 : T$) {
  - FOR($i = 1 : N_s$) {
    - IF($t = 1$) $\alpha_t^i \sim \mathcal{N}_1(\alpha_1^i|\mu_1, \sigma_1)$
      ELSE $\alpha_t^i \sim \mathcal{N}_1(\alpha_t^i|\alpha_{t-1}^i + \delta_\alpha, \sigma_\delta)$
    - $w_t^i = P(\boldsymbol{o}_t|\alpha_t^i)$ }
  - normalize :w_t^i for $\sum_{i=1}^{N_s} w_t^i = 1$
  - resampling using following algorithm }

Resampling

- $c_1 = 0$, FOR($i = 2 : N_s$) { $c_i = c_{i-1} + w_t^i$ }
- $i = 1$, $u_1 \sim$ Uniform Distribution of $[0, N_s^{-1}]$
- FOR($j = 1 : N_s$) {
  - $u_j = u_1 + N_s^{-1}(j-1)$
  - while $u_j > c_i$, $i = i + 1$
  - $\alpha_t^j = \alpha_t^i$, $w_t^j = N_s^{-1}$ }

Figure 2: Particle filter algorithm.

4.2 M-step

By solving the stationary condition $\partial L / \partial \boldsymbol{\theta} = 0$, the maximizers of $L(\boldsymbol{\theta})$ is given by

$$\mu_1 = \langle \alpha_1 \rangle, \quad \delta_\alpha = \frac{\sum_{t=2}^{T} (\langle \alpha_t \rangle - \langle \alpha_{t-1} \rangle)}{T-1},$$

$$\boldsymbol{v}' = \left(\sum_{t=1}^{T} \boldsymbol{o}_t' \langle \boldsymbol{B}(\alpha_t) \rangle \right) \left(\sum_{t=1}^{T} \langle \boldsymbol{B}'(\alpha_t) \boldsymbol{B}(\alpha_t) \rangle \right)^{-1},$$

where $\langle \cdot \rangle$ is expectation under $Q(A_1^T)$, calculated by

$$\langle \alpha_t \rangle = \frac{1}{N_s} \sum_{i=1}^{N_s} (\alpha_t^i), \quad \langle \boldsymbol{B}(\alpha_t) \rangle = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{B}(\alpha_t^i),$$

$$\langle \boldsymbol{B}'(\alpha_t) \boldsymbol{B}(\alpha_t) \rangle = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{B}'(\alpha_t^i) \boldsymbol{B}(\alpha_t^i).$$

The maximizers are substituted for $\bar{\boldsymbol{\theta}}$.

5 Experiment

We evaluated our method using two datasets. One is a synthetic dataset generated by the probabilistic model (1). The other is a real dataset, which was obtained by running a Khepera robot in a real environment. Both datasets were generated by the following procedure. The initial state was set at $\alpha_1 = 0$,

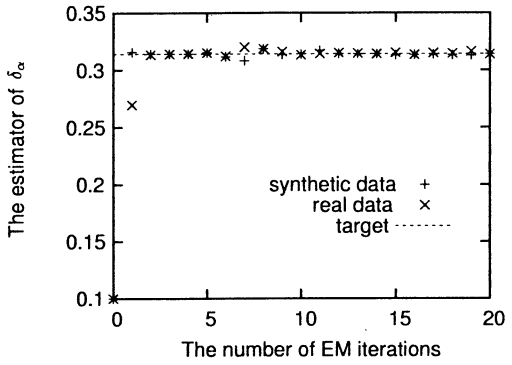


Figure 3: The estimators of δ_α.

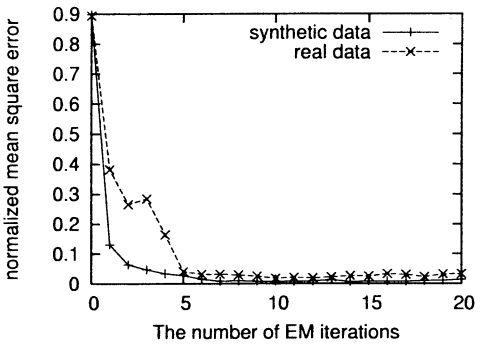


Figure 4: Inaccuracy of the estimated observation models.

and then the robot rotated at the angle of $\delta_\alpha = \pi/10$ per unit action. A series of actions continued until $T = 40$. Namely, observation data was generated by turning the robot twice. The constant parameters corresponding to the noise level were set at $(\sigma_1, \sigma_\delta, \sigma_o) = (1.0, 2.0, 500)$ [1]. The estimators of unknown parameters were initialized by $(\bar{\mu_1}, \bar{\delta_\alpha}) = (0, 0.1)$, while $\bar{v}$ was initialized at random.

Figure 3 shows the time courses of the estimator for δ_α. The dash line indicates the true value. Marks "+" and "×" indicate the estimators obtained from the synthetic and real datasets, respectively. The estimator for μ_1 was changed little at each EM iteration and immediately converged to 0 with a small error. These results show that the state transition processes for both datasets were well identified in a few EM iterations.

Since the true value of $\boldsymbol{v}$ was unknown in a real dataset, we evaluated the accuracy of the observation process as follows. According to the probabilistic model (1) with the estimators $\bar{\boldsymbol{\theta}}$, we generated the virtual observation, which is called the reproduced observation $\bar{\boldsymbol{O}}_1^T$. We then calculated the mean square error between the true observation $\boldsymbol{O}_1^T$ and the reproduced one $\bar{\boldsymbol{O}}_1^T$, which was normalized by the variance of $\boldsymbol{O}_1^T$. We used the normalized mean square error (nMSE) as the criterion of the inaccuracy. Figure 4 shows the times courses of the nMSE. The errors for both datasets decreased rapidly and converged to a very small value in about ten EM iterations. Although this analysis was not enough to evaluate the accuracy, it was suggested that the estimated models are consistent with the true observation processes.

[1] In generating the synthetic dataset, the parameters were set at $(\sigma_1, \sigma_\delta, \sigma_o) = (0.0001, 0.0001, 0.0001)$. Namely, the true process was disturbed by a very small noise.

6 Conclusion and discussion

In this study, we proposed a method to identify state transition and observation processes. We assumed that the parameters corresponding to the noise level were given but they are unknown in general. Although our method can be modified so as to estimate these parameters, we have found that the estimators tended to be larger than the true values. One possible reason for this problem is that the particle filter algorithm is not enough to approximate the true posterior distribution of unknown variables. We will modify a sampling distribution in the algorithm in order to get the approximation better in our future work.

Although we restricted the robot's action to the rotation, we should consider other actions such as "going forward" and "turning left" in our future work. In order to deal with such a situation, we will also extend the state space of our method to three dimensions.

References

[1] S. Thrun, W. Burgard, and D. Fox (1998), "A probabilistic approach to concurrent mapping and localization for mobile robots", Machine Learning, 31(5), pp.1-25.

[2] S. Roweis, and Z. Ghahramani (2001), "An EM algorithm for identification of nonlinear dynamical systems", In S. Haykin, editor, Kalman Filtering and Neural Networks, pp.175-220. Wiley, New York.

[3] S. Arulampalam, S. Maskell, N. Gordon, et al. (2002), "A Tutorial on Particle Filters for Online Non-linear/Non-Gaussian Bayesian Tracking", IEEE Transactions on Signal Processing, 50(2), pp.174-188.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

An Approximate Stability Analysis of a Robust Control DC Motor System Using Higher Order Derivatives of Universal Learning Networks

Ahmed Hussein
Kyushu University
Fukuoka, Japan

Kotaro Hirasawa
Waseda University
Kitakyushu, Japan

Jinglu Hu
Waseda University
Kitakyushu, Japan

Kiyoshi Wada
Kyushu University
Fukuoka, Japan

Abstract

As for the stability of nonlinear dynamical systems, Lyapunov's direct method has been widely used. But finding an appropriate Lyapunov function is fairly difficult especially for complex nonlinear dynamical systems. Therefore, in this paper a new stability analysis method based on Higher Order Derivatives (HODs) of Universal Learning Networks (ULNs) is described, and its application to a separately excited robust dc motor fed from a photovoltaic generator and loaded with a constant torque is discussed.

1 Introduction

Studying the stability of dynamical systems goes back to 1868 when J.C. Maxwll studied the stability of the systems by linearizing the differential equations and checking the solutions of their characteristic polynomial expressions [1]. In 1898, A.M. Lyapunov established his direct method which is very general one based on the energy concept in Lagrange's preserved systems [2]. In addition, the construction procedures of the Lyapunov function were extensively studied by Krasovski [3]. But it is still hard to find an appropriate Lyapunov function for nonlinear dynamical systems. On the other hand, many studies on the dc motor control system design methods, especially by using neural networks, have been done recently [4-6], but they did not concern with the robustness or/and the stability problems. Furthermore, in our earlier work [7], the robustness of a separately excited dc motor fed from Photovoltaic (PV) generator and loaded with a constant torque is realized by the Higher Order Derivatives (HODs) of Universal Learning Networks (ULNs). Therefore, as an extension of the research given in [7] and to overcome the above mentioned problems, in this paper a new stability analysis method using the HODs of ULNs [8] is proposed. In this method, the stability of the original trajectory of the robust dc motor system is studied by checking whether the perturbed trajectory caused by any disturbance, can approach the original trajectory or not. Trajectory here means the dynamics

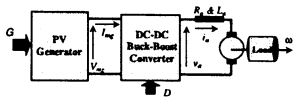


Figure 1: The proposed dc motor system

of the system from an initial state to a final state. By investigating the first order derivatives of the original trajectory, we can easily determine whether the original trajectory is locally asymptotically stable or not. This is another approach for the linearized stability analysis, which does not require the complex eigenvalue calculations. In our simulations, the above analysis can be achieved by calculating the first order derivatives of any node of the trajectory w.r.t. the initial change of the load torque. That is, if they approach zero at time infinity, then the trajectory is locally asymptotically stable.

This paper is organized as follows: In the next section, the mathematical model of the main devices forming the dc motor system is given. A new scheme of stability analysis using HODs of ULNs is introduced in section 3. In section 4, simulation results of the proposed stability method of the robust dc motor system are presented. Finally, section 5 is devoted to the conclusions.

2 DC Motor System

The dc motor system consists of three main devices: PV generator, dc-dc converter and dc motor loaded with constant torque as shown in Fig. 1.

2.1 PV Generator Model

The PV generator consists of solar cells connected in series and parallel fashion to provide the desired voltage and current required by the system. This PV generator exhibits a nonlinear voltage-current characteristics governed by Eq. (1).

$$V_g = \frac{1}{\Lambda_g} \ln\left(\frac{G \times I_{phg} + I_{og} - I_g}{I_{og}}\right) - I_g R_{sg} \qquad (1)$$

Where V_g is PV generator voltage; I_g is PV generator current; Λ_g is PV generator constant; R_{sg} is PV generator series resistance; I_{phg} is insolation-dependent photo current of the PV generator; I_{og} is PV generator reverse saturation current; G is per unit solar insolation. From Eq. (1), at any value of G there exists only one point at which the output power is maximum. This point is called the Maximum Power Point (MPP). Due to the high cost of the PV generators, it is recommended to operate at this MPP at all values of G.

2.2 DC Motor Model

The dynamics of the separately excited dc motor and its load are represented by these two equations:

$$v_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K\omega(t) \quad (2)$$

$$Ki_a(t) = B\omega(t) + J_a \frac{d\omega(t)}{dt} + T_L \quad (3)$$

Where v_a is motor voltage; i_a is motor current; ω is motor speed; J_a is motor shaft inertia; R_a is armature resistance; L_a is armature inductance; K is torque and back EMF constant; B is motor damping; T_L is load torque. Using a sampling time interval ΔT of 0.001s, and a first and a second-order, finite-backward difference approximation for $\frac{d\omega}{dt}$ and $\frac{d^2\omega}{dt^2}$, respectively, the finite difference equation that governs the discrete-time dynamics of the dc motor is given by Eq. (4).

$$\omega(t) = \alpha \frac{V_{mg}(t)D(t)}{1 - D(t)} + \beta\omega(t-1) + \delta\omega(t-2) + \theta T_L \quad (4)$$

Where D is duty ratio of the dc-dc converter. α, β, δ, and θ are constants depending on ΔT and the motor parameters.

2.3 DC-DC Converter Model

The dc-dc converter is inserted between the PV generator and the dc motor to match the PV output characteristics to the motor input characteristics. The most important parameter of the converter is its chopping ratio Y that depends on the duty ratio D through a nonlinear relation, $Y(t) = \frac{D(t)}{1-D(t)}$. Assuming the converter is ideal, then its input and output powers are equal resulting in the following relation:

$$\frac{v_a(t)}{V_{mg}(t)} = \frac{I_{mg}(t)}{i_a(t)} = Y(t) \quad (5)$$

Where V_{mg} and I_{mg} is the voltage and current corresponding to the MPP, respectively.

3 Stability Analysis by ULNs

The stability of the trajectory is generally defined as follows [9]: For arbitrary positive real number ε and arbitrary time t_0, the original trajectory of ULNs is stable if there exists $\delta(\varepsilon, t_0)$ such that

$$\| \Delta h(t_0) \| < \delta \quad \text{and} \quad \| \Delta h(t) \| < \varepsilon \quad \text{for all} \quad t \geq t_0,$$

Otherwise the original trajectory is unstable. Where $\Delta h(t_0) = (\Delta h_1(t_0), \cdots, \Delta h_{|J|}(t_0))$ is the initial disturbance vector subjected to the original trajectory; $\Delta h(t) = (\Delta h_1(t), \cdots, \Delta h_{|J|}(t))$ is the orbital change vector between the original trajectory and the perturbed trajectory at time t caused by $\Delta h(t_0)$.
On the other hand, calculating Taylor expansion of the original trajectory, we can obtain $\| \Delta h(t) \|$ as follows:

$$\| \Delta h(t) \| = \sqrt{\sum_{r \in J} \Delta h_r(t)^2} \quad (6)$$

$$\Delta h_r(t) \approx \sum_{r1 \in J} \frac{\partial^\dagger h_r(t)}{\partial h_{r1}(t_0)} \times \Delta h_{r1}(t_0) + \frac{1}{2} \sum_{r1 \in J} \sum_{r2 \in J} \frac{\partial^{\dagger 2} h_r(t)}{\partial h_{r1}(t_0) \partial h_{r2}(t_0)} \times \Delta h_{r1}(t_0) \Delta h_{r2}(t_0) + \cdots + \frac{1}{n!} \sum_{r1 \in J} \cdots \sum_{rn \in J} \frac{\partial^{\dagger n} h_r(t)}{\partial h_{r1}(t_0) \cdots \partial h_{rn}(t_0)} \times \Delta h_{r1}(t_0) \cdots \Delta h_{rn}(t_0). \quad (7)$$

Where h_r is the output of node r and J is the set of suffixes of the nodes. Therefore, the perturbed trajectory needed for studying the stability of the nonlinear systems can be easily calculated by using the first order derivatives of ULNs instead of calculating it directly at every initial disturbance $\Delta h(t_0)$. The proposed method is essentially the same as conventional linearized ones. But the iterative calculation for obtaining the first order derivatives is easy to handle compared to the conventional ones requiring the calculation of eigenvalues of Jacobians. Furthermore, the iterative calculation does not need to solve high order algebraic equations at all. From Eq. (7), it is observed that the linear deviation of the trajectory calculated by $\sum_{r1 \in J} \frac{\partial^\dagger h_r(t)}{\partial h_{r1}(t_0)} \times \Delta h_{r1}(t_0)$ can be obtained by the first order derivative. Let us introduce the notation $P_1(j, t, h_{r1}(t_0))$ to represent the first order derivative $\frac{\partial^\dagger h_j(t)}{\partial h_{r1}(t_0)}$ whose calculation requires the direct and indirect relation between h_j and h_{r1} as given by the following forward propagation algorithm [10]:

$$P_1(j, t, h_{r1}(t_0)) = \sum_{i \in J_F(j)} \Big[\frac{\partial h_j(t)}{\partial h_i(t - \tau_{ij})} \times P_1(i, t - \tau_{ij}, h_{r1}(t_0)) \Big] + \frac{\partial h_j(t)}{\partial h_{r1}(t_0)}, \; j \in J, \; t \in T \quad (8)$$

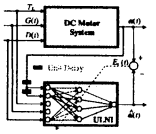


Figure 2: Identification scheme of the dc motor system

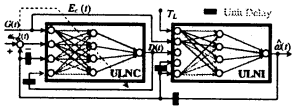


Figure 3: Control scheme of the dc motor system

Where, $h_j(t)$ is the output of node j at time t; $J_F(j)$ is the set of suffixes of nodes whose outputs are connected to node j; T is the set of sampling instants and τ_{ij} is the time delay of the branch from node i to node j. If the branch connecting nodes i to j has no time delay, then τ_{ij} is set to zero. The initial values of $\frac{\partial h_j(t)}{\partial h_{r1}(t_0)}$ are set to 1.0 when $j = r_1$ and $t = t_0$. Otherwise their values are set to zero. Therefore, the locally asymptotical stability of the nonlinear trajectory can be studied as follows:

$$\text{If } \lim_{t\to\infty} \left| \frac{\partial^\dagger h_r(t)}{\partial h_{r1}(t_0)} \right| = 0 \quad \text{for any } r, r_1 \in J,$$

Then the trajectory becomes locally asymptotically stable, because

$$\lim_{t\to\infty} \| \Delta h(t) \| = 0 \quad \text{for any small} \quad \| \Delta h(t_0) \| .$$

4 Simulations of Stability of Robust DC Motor System

Before studying the stability, the dc motor system is identified offline by using the Universal Learning Network Identifier (ULNI) as shown in Fig.2. The ULNI is modelled as a FFNN with an architecture of 5-10-1. The output of the ULNI is the identified motor speed $\hat{\omega}(t)$. The inputs $D(t)$ and $G(t)$ are generated randomly considering that they must cover the expected range of the real input signals. The load torque T_L is set to 20 N.m. After the identification process is completed, the dc motor system is replaced by the ULNI and connected in series with the Universal Learning Network Controller (ULNC) as shown in Fig. 3 for the online training of the robust controller. The main objective of the ULNC is to control $D(t)$ continuously, so that the motor can operate at the MPP of the PV source and its speed can follow a reference signal $\omega_{ref}(t)$. The type of the network used as a ULNC is the same as that used as an identifier with an architecture of 4-10-1. The aim of the robust ULNC is to restrain the effect caused by the load torque changes. In other words, the dc motor system must be controlled so that the difference between the system behavior at the learning stage and that at the control stage is minimized. To realize this, the criterion function L of the ULNC containing two parts is introduced; Fundamental part E_c and extended part E_x such that $L = E_c + \gamma E_x$, where γ is the robustness coefficient. For our application, E_c and E_x are given as follows:

$$E_c = \tfrac{1}{2} \sum_{s\in T_c} \left[\omega_{ref}(s) - \hat{\omega}(s)\right]^2, \; E_x = \left[\frac{\partial^\dagger E_c}{\partial T_L}\right]^2$$

where T_c is the set of updating window size instants of the ULNC. The performance of the ULNC is investigated for a reference speed ω_{ref} and solar insolation G given in Eq.(9) and Eq.(10), respectively.

$$\omega_{ref}(s) = 110 + 15[sin(2\pi s\Delta T) + cos(2\pi s\Delta T)] \quad (9)$$

$$G(s) = 0.3 + 0.7\ sin(\pi s\Delta T) \quad (10)$$

The controller free parameter λ_c is updated every $\mid T_c \mid$ sampling instants based on the gradient method given by (11), where η_c and μ_c are the learning rate and the momentum coefficient of the ULNC, respectively.

$$\lambda_c \leftarrow \lambda_c - \eta_c \frac{\partial^\dagger L}{\partial \lambda_c} + \mu_c \Delta\lambda_c \quad (11)$$

The stability method described in section 3 is explained by studying the locally asymptotical stability of the original trajectory of the dc motor system using ULNs. This is done by calculating the first order derivatives of the controller output $\hat{\omega}(t)$ w.r.t. the initial changes in the load torque T_L and checking their convergence supposing that an initial disturbance is given to T_L at time t_0. Based on the discussion in section 3, if these first order derivatives approach zero at the time infinity, then the trajectory is locally asymptotically stable. In this paper, the proposed method is investigated at three different values of T_L; that is 15, 20 and 25 N.m. Also the effectiveness of the proposed stability method is studied at two different values of the robustness coefficient $\gamma = 100$ and $\gamma = 0$. In other words, the stability analysis method is discussed when the robustness of the dc motor system is/is not considered. Therefore the ULNC is trained online at T_L=15, 20 and 25 N.m. The corresponding values of η_c and μ_c are adjusted to 10^{-4} & 0.008, 10^{-4} & 0.01 and 10^{-5} & 0.07, respectively. Once the training procedure is completed, we calculate the first order derivatives $\frac{\partial^\dagger h_j(t)}{\partial T_L(t_0)}$ by replacing $h_{r1}(t_0)$ with $T_L(t_0)$ in Eq. (8) supposing that T_L is a function of sampling instants as follows:

$$\frac{\partial^\dagger h_j(t)}{\partial T_L(t_0)} = \sum_{i\in J_F(j)} \left[\frac{\partial h_j(t)}{\partial h_i(t-\tau_{ij})} P_1(i, t-\tau_{ij}, T_L(t_0))\right] + \frac{\partial h_j(t)}{\partial T_L(t_0)}, \qquad j \in J, \; t \in T \quad (12)$$

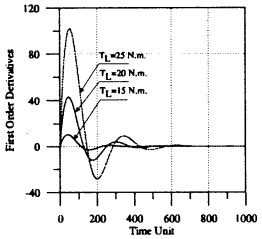


Figure 4: First order derivatives at $\gamma = 0$

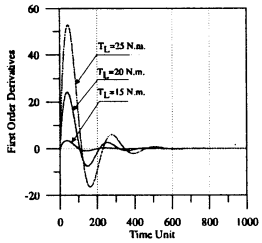


Figure 5: First order derivatives at $\gamma = 100$

For the node j whose input is T_L, the term $\frac{\partial h_j(t)}{\partial T_L(t_0)}$ has a unity value at $t = t_0$ only. Furthermore, the initial values of the first order derivatives $P_1(i, t_0, T_L(t_0))$ are set to zero.
The first order derivatives $\frac{\partial^\dagger \tilde{\omega}(t)}{\partial T_L(t_0)}$ using $\gamma = 0$ and T_L = 15, 20 and 25 N.m. are shown in Fig.4.
To demonstrate the effectiveness of the robust ULNC, the simulation is repeated for $\gamma = 100$ as shown in Fig.5. From these figures, it is obvious that all of the first order derivatives curves reached zero at the final state. This means, all the trajectories are locally asymptotically stable. But the transient of each curve is different from others depending on the load torque T_L. Also we can notice that the maximum amplitude of the trajectory at $T_L = 25$ N.m. has the highest value, while the maximum amplitude of the trajectory at $T_L = 15$ N.m. has the lowest value. This means the motor operation at the later case is more stable than that at the former case. In that sense, comparing Fig. 4 and 5 based on the maximum amplitude of the trajectories, we can obtain that the motor operation at $\gamma = 100$ is more stable than that at $\gamma = 0$.

5 Summary

In this paper, a new stability analysis method based on the HODs of ULNs is discussed. Also its application to a separately excited dc robust motor loaded with a constant torque and fed from PV generator via dc-dc converter is explained. In this method, stability of the dc motor system is studied by calculating the first order derivatives of any node of the trajectory w.r.t. the initial disturbance $T_L(t_0)$ and checking if they approach zero as time approaches infinity. If they approach zero, then the system is locally asymptotically stable. The simulation results showed that as T_L increased, the system tendency to instability is also increased. Furthermore, it was also clarified that the robust dc motor system is more stable than without robustness.

Acknowledgements

This research was partly supported by the 21^{st} Century COE Program "Reconstruction of Social Infrastructure Related to Information Science and Electrical Engineering".

References

[1] J.C. Maxwell, "On Governers", *Proc. R. Soc. London*, Vol. 16, 1868.

[2] A.M. Lyapunov, "Stability of Motion", *Academic Press, 1898.*

[3] N.N. Krasovski, "On the Stability in the Large of a System of Nonlinear Differential Equations", *Prikl. Math. i Mekh.*, Vol. 18, 1954.

[4] S. Weerasooriya and M.A. El-Sharkawi, "Identification and Control of a DC Motor Using Back Propagation Neural Networks", *IEEE Trans. Energy Conversion*, Vol. 6, pp. 663-669, 1991.

[5] L. da Silva, *et.al.*, "Simulation of a Neural Net Controller for Motor Drives", *Mathematics and Computers in Simulation*, Vol. 38, pp. 311-322, 1995.

[6] A. Rubaai and R. Kotaru, "Online Identification and Control of a DC Motor Using Learning Adaptation of Neural networks", *IEEE Trans. Industry Application*, Vol. 36, pp. 935-942, 2000.

[7] A. Hussein, K. Hirasawa and J. Hu, "Robust Control of a PV-Supplied DC Motor Using Higher Order Derivatives of ULNs", *SICE Annual Conference in Fukui*, Fukui University, Japan, pp. 26-31, August 4-6, 2003.

[8] K. Hirasawa, J. Murata, J. Hu and C. Jin, "Universal Learning Network and Its Application to Robust Control", *IEEE Trans. System Man and Cybernetics-part B*, Vol. 3, No. 3, pp. 419-430, 2000.

[9] J.P. Lasalle, "The Stability and Control of Discrete Processes", *Springer-Verlag*, 1986.

[10] K. Hirasawa, X. Wang, J. Murata, J. Hu and C. Jin, "Universal Learning Network and Its Application to Chaos Control", *Neural Networks*, Vol. 13, pp. 239-253, 2000.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Experiment of Digital RAC for an Underwater Robot with Vertical Planar 2-Link Manipulator

Shinichi Sagara Kenzo Shibuya Masakazu Tamura
Department of Control Engineering, Kyushu Institute of Technology
Tobata, Kitakyushu 804-8550, Japan
E-mail: sagara@cntl.kyutech.ac.jp

Abstract

Through experiments of an underwater robot with 2 dimensional and vertical planar 2-link manipulator, we have shown that a Resolved Acceleration Control (RAC) with the position and velocity feedback has a good control performance in spite of using an inaccurate hydrodynamic model. We have also proposed a digital RAC method. In this paper, the effectiveness of the digital version is verified by experiment.

1 Introduction

Since underwater robots are necessary for ocean exploration and preservation, many studies have been done about Underwater Robotic Vehicles (URVs) [1]. Recent years, dynamics and control of Underwater Vehicle-Manipulator Systems (UVMSs) have been studied because the manipulators are used for collecting resources, repairing cables, constructing structures, etc [2, 3, 4, 5]. However, the experimental studies of UVMS are only a few.

Through experiments of a free floating underwater robot with horizontal planar 2-link manipulator, we have shown that a Resolved Acceleration Control (RAC) method with the manipulator's end-tip position and velocity feedback has good control performance [6]. Furthermore, for an underwater robot with vertical planar 2-link manipulator shown in Fig. 1, we have done the position control experiment both the robot base and the end-tip of manipulator using RAC method [7].

In general, digital computers are utilized for URV's controller and a few digital control methods have been proposed [8]. We have also proposed a design of digital RAC and shown that the control performance is similar to the analog version [9].

In this paper, we show the experimental result of the underwater robot with vertical planar 2-link manipulator using the digital RAC method. The experimental result shows that the digital control method has a good control performance in spite of incorrect hydrodynamic force.

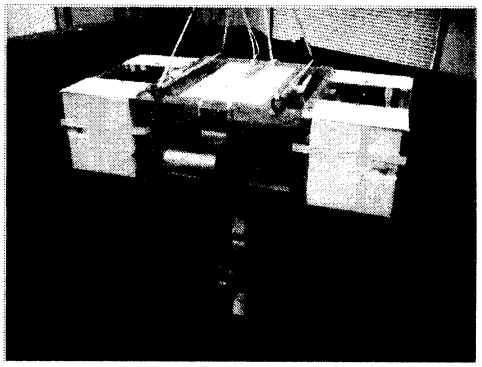


Fig. 1: Floating 2-link underwater robot

2 Modeling [7]

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 vertical plane. Thrusters are mounted on the base to provide propulsion for position and attitude control of the base.

Symbols used in this paper are defined as follows:

- Σ_U : inertial coordinate frame
- Σ_i : ith link coordinate frame ($i = 0, 1, 2$; link 0 means base)
- ${}^U\boldsymbol{R}_i$: coordinate transformation matrix from Σ_i to Σ_U
- l_i : length of link i
- $\boldsymbol{v}_i$: velocity vector of link i with respect to Σ_i
- ϕ_i : relative joint angle
- $\boldsymbol{p}_0$: position vector of origin of Σ_o with respect to Σ_U
- $\boldsymbol{p}_e$: position vector of end-tip with respect to Σ_U
- $\boldsymbol{x}_0$: position and attitude vector of base with respect to Σ_U ($= [\boldsymbol{p}_0^T,\ \theta_0]^T$)
- $\boldsymbol{\phi}$: joint angle vector ($= [\phi_1,\ \phi_2]^T$)

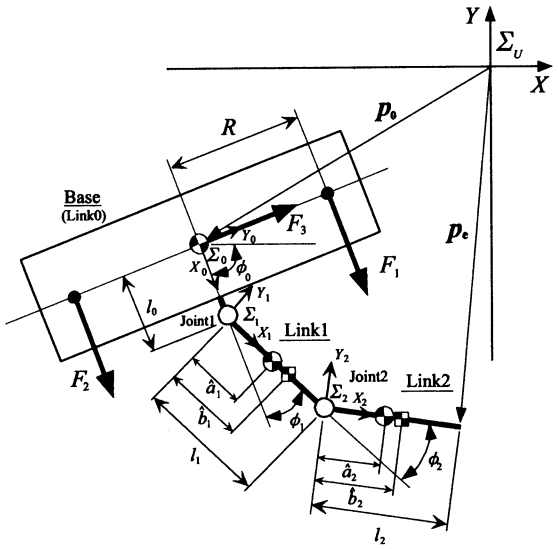


Fig. 2: 2-link underwater robot model

m_i : mass of link i (link 0 means the robot base)
$\boldsymbol{I}_i$: inertia tensor of link i
$\hat{\boldsymbol{x}}_i$: position vector from joint i to joint $(i+1)$ with respect to Σ_i
$\hat{\boldsymbol{a}}_i$: position vector from joint i to center of gravity of link i with respect to Σ_i
$\hat{\boldsymbol{b}}_i$: position vector from joint i to center of buoyancy of link i with respect to Σ_i
$\boldsymbol{E}$: unit matrix
F_i : thruster force $(i = 1,\ 2,\ 3)$
R : length form origin of Σ_o to thruster

2.1 Kinematics

First, a time derivative of the end-tip position vector $\boldsymbol{p}_e$ is

$$\dot{\boldsymbol{p}}_e = \boldsymbol{A}\dot{\boldsymbol{x}}_0 + \boldsymbol{B}\dot{\boldsymbol{\phi}} \tag{1}$$

where $\boldsymbol{A} \in R^{2\times 3}$ and $\boldsymbol{B} \in R^{2\times 2}$ are matrices consisting of attitude angle of base and joint angles.

Next, let $\boldsymbol{\eta}$ and $\boldsymbol{\mu}$ be a linear and an angular momentum of the robot including hydrodynamic added mass tensor $\boldsymbol{M}_{a_i}$ and added inertia tensor $\boldsymbol{I}_{a_i}$ of link i, then

$$\boldsymbol{\eta} = [\eta_1,\ \eta_2,\ 0]^T = \sum_{i=0}^{2} {}^U\boldsymbol{R}_i(m_i\boldsymbol{E} + \boldsymbol{M}_{a_i})\tilde{\boldsymbol{a}}_i, \tag{2}$$

$$\boldsymbol{\mu} = [0,\ 0,\ \mu_3]^T = \sum_{i=0}^{2}(\boldsymbol{I}_i + \boldsymbol{I}_{a_i})\boldsymbol{\omega}_i + \hat{\boldsymbol{x}}_i \times \{{}^U\boldsymbol{R}_i(m_i\boldsymbol{E} + \boldsymbol{M}_{a_i})\tilde{\boldsymbol{a}}_i\} \tag{3}$$

where $\tilde{\boldsymbol{a}}_i = \boldsymbol{v}_i + \boldsymbol{\omega}_i \times \hat{\boldsymbol{a}}_i$ and $\boldsymbol{\omega}_i = [0,\ 0,\ \dot{\phi}_i]^T$.

From Eqs. (1), (2) and (3) the following equation can be obtained:

$$\boldsymbol{s} = [\eta_1,\ \eta_2,\ \mu_3]^T = \boldsymbol{C}\dot{\boldsymbol{x}}_0 + \boldsymbol{D}\dot{\boldsymbol{\phi}} \tag{4}$$

where $\boldsymbol{C} \in R^{3\times 3}$ and $\boldsymbol{D} \in R^{3\times 2}$ are matrices including the added mass $\boldsymbol{M}_{a_i}$ and the added inertia $\boldsymbol{I}_{a_i}$.

2.2 Hydrodynamic force and moment

Generally, the drag force and moment of the joint i can be represented as follows [10, 11]:

$$\boldsymbol{f}_{d_i} = \frac{\rho}{2}C_{D_i}D_i\int_0^{l_i} ||\boldsymbol{w}_i||\boldsymbol{w}_i d\hat{\boldsymbol{x}}_i, \tag{5}$$

$$t_{d_i} = \frac{\rho}{2}C_{D_i}D_i\int_0^{l_i} \hat{\boldsymbol{x}}_i||\boldsymbol{w}_i||\boldsymbol{w}_i d\hat{\boldsymbol{x}}_i \tag{6}$$

where $\boldsymbol{w}_i = \boldsymbol{v}_i + \boldsymbol{\omega}_i \times \hat{\boldsymbol{x}}_i$, and ρ is the fluid density, C_{D_i} is the drag coefficient, D_i is the width of link i.

The gravitational and buoyant forces acting link i are described as follows:

$$\boldsymbol{f}_{g_i} = ({}^U\boldsymbol{R}_i)^T(\rho V_i - m_i)\boldsymbol{g}, \tag{7}$$

$$\boldsymbol{t}_{g_i} = ({}^U\boldsymbol{R}_i)^T(\hat{\boldsymbol{b}}_i \times \rho V_i\boldsymbol{g} - \hat{\boldsymbol{a}}_i \times m_i\boldsymbol{g}) \tag{8}$$

where V_i is the volume of link i and $\boldsymbol{g}$ is the gravitational acceleration vector.

3 Digital RAC [9]

Discretizing the time differentiated Eqs. (1) and (4) by a sampling period T, the following equation can be obtained:

$$\boldsymbol{W}(k)\boldsymbol{\alpha}(k-1) = \frac{1}{T}[\boldsymbol{\nu}(k) - \boldsymbol{\nu}(k-1) + T\boldsymbol{f}(k) - \{\boldsymbol{W}(k) - \boldsymbol{W}(k-1)\}\boldsymbol{v}(k)] \tag{9}$$

where

$$\boldsymbol{W} = \begin{bmatrix}\boldsymbol{H} & \boldsymbol{D}\\ \boldsymbol{A} & \boldsymbol{B}\end{bmatrix},\quad \boldsymbol{H} = \boldsymbol{C} + \boldsymbol{E},\quad \boldsymbol{\nu} = \begin{bmatrix}\dot{\boldsymbol{x}}_0\\ \dot{\boldsymbol{p}}_e\end{bmatrix},$$

$$\boldsymbol{\alpha} = \left[\ddot{\boldsymbol{x}}_0^T,\ \ddot{\boldsymbol{\phi}}^T\right],\quad \boldsymbol{v} = \left[\dot{\boldsymbol{x}}_0^T,\ \dot{\boldsymbol{\phi}}^T\right],\quad \boldsymbol{f} = [\dot{\boldsymbol{s}}^T,\ \boldsymbol{0}]$$

and $\dot{\boldsymbol{s}}$ is the external force including hydrodynamic force and thrust of the thruster which act on the robot. Note that computational time delay is introduced to Eq. (9) and the discrete time kT is abbreviated to k.

For Eq. (9), the desired acceleration is defined as follows:

$$\boldsymbol{\alpha}_d(k) = \frac{1}{T}\boldsymbol{W}^{-1}(k)[\boldsymbol{\nu}_d(k+1) - \boldsymbol{\nu}_d(k) + \boldsymbol{\Lambda}\boldsymbol{e}_\nu(k) + \boldsymbol{\Gamma}\boldsymbol{e}_p(k) + T\boldsymbol{f}(k)] \tag{10}$$

where $\boldsymbol{e}_\nu(k) = \boldsymbol{\nu}_d(k) - \boldsymbol{\nu}(k)$ and $\boldsymbol{e}_p(k) = \boldsymbol{p}_d(k) - \boldsymbol{p}(k)$, $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_i\}$ $(i = 1, \cdots, 5)$, $\boldsymbol{\Gamma} = \mathrm{diag}\{\gamma_i\}$, $\boldsymbol{p}(k) = [\boldsymbol{x}_0^T, \ \boldsymbol{p}_e^T]^T$, $\boldsymbol{\nu}_d(k)$ and $\boldsymbol{p}_d(k)$ are the desired values of $\boldsymbol{\nu}(k)$ and $\boldsymbol{p}(k)$.

Assuming $\boldsymbol{W}(k) \approx \boldsymbol{W}(k-1)$ and $\boldsymbol{f}(k) \approx \boldsymbol{f}(k-1)$ for one sampling period, from Eqs. (9) and (10) we have

$$\boldsymbol{W}(k)\boldsymbol{e}_\alpha(k) = \frac{1}{T}\{[(q-1)\boldsymbol{E} + \boldsymbol{\Lambda}]\boldsymbol{e}_\nu(k) + \boldsymbol{\Gamma}\boldsymbol{e}_p(k)\} \quad (11)$$

where $\boldsymbol{e}_\alpha(k) = \boldsymbol{\alpha}_d(k) - \boldsymbol{\alpha}(k)$ and q is the forward sift operator. Furthermore, applying $\boldsymbol{e}_\nu(k)$ to the backward Euler approximation, and selecting $\gamma_i = (2 - \lambda_i - 2\sqrt{1-\lambda_i})/T$, the following equation can be obtained:

$$\boldsymbol{W}(k)\boldsymbol{e}_\alpha(k) = \frac{1}{T^2} q^{-1}(q\boldsymbol{E} - \tilde{\boldsymbol{\Lambda}})^2 \boldsymbol{e}_p(k) \quad (12)$$

where $\tilde{\boldsymbol{\Lambda}} = \mathrm{diag}\{\sqrt{1-\lambda_i}\}$. Therefore, if the feedback gain λ_i is selected to satisfy $0 < \lambda_i < 1$, $\boldsymbol{e}_\alpha(k) \to \boldsymbol{0}$ $(k \to \infty)$ and all elements of $\boldsymbol{W}(k)$ are bounded, $\boldsymbol{e}_p(k) \to \boldsymbol{0}$ $(k \to \infty)$ can be ensured.

4 Experiment

In this section, to verify the effectiveness of the digital RAC method, the experiment is done.

4.1 Experimental system

Fig. 3 shows a configuration of experimental system. A robot has a 2DOF manipulator that joints are actively rotated by a velocity control type servo actuator consisting of DC servo motor and incremental type encoder. Four thrusters are attached to vertical and horizontal directions on the robot base. Physical parameters of the underwater robot is shown in Table 1. And the dynamic equation of the robot is shown in [7].

Measurement and control system are consist of CCD camera, Video Tracker and 32bit personal computer. Two light emitting diodes (LED) is attached to the base, and its motion is monitored by a CCD camera. Video signals of the LED markers are transformed into the position data by XY-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.

40[W] thrusters which provide propulsion for controlling the position and attitude angle of the base are

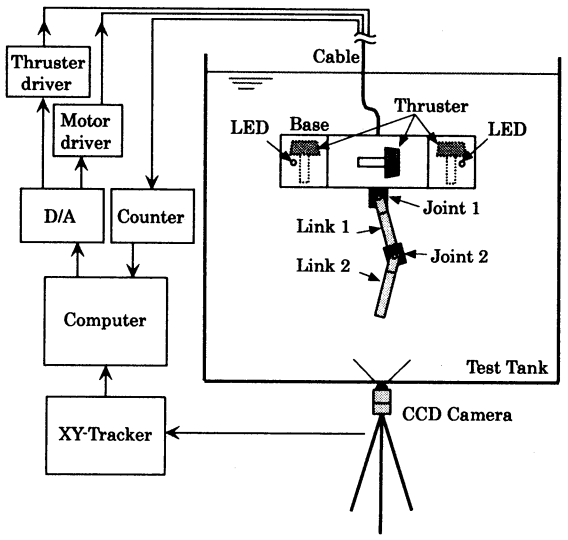


Fig. 3: Configuration of underwater robot system

utilized. Static characteristics of the thruster is obtained by experiment [7]:

$$F_{CW} = 1.341v^2 - 1.363v - 0.026, \quad (13)$$

$$F_{CCW} = 0.763v^2 - 8.345v - 0.019 \quad (14)$$

where v is the input voltage to the power amplifier of the thruster.

4.2 Experimental result

The experiment was carried out under the following condition. The desired end-tip position was set up along a straight path from the initial position to the target calculated from trapezoidal velocity pattern. On the other hand, the desired position and attitude of base were set up the initial values. The sampling period was $T = 1/60[s]$ and the feedback gains were $\lambda_1 = 0.4$, $\lambda_2 = 0.4$, $\lambda_3 = 0.3$, $\lambda_4 = 0.4$ and

Table 1: Physical parameters of the underwater robot

	Base	Link 1	Link 2
Mass [kg]	26.04	4.25	1.23
Moment of inertia [kg m^2]	1.33	0.19	0.012
Link length (x axis) [m]	0.2	0.25	0.25
Link length (y axis) [m]	0.81	0.04	0.04
Link width [m]	0.42	0.12	0.12
Added mass(x) [kg]	72.7	1.31	0.1
Added mass(y) [kg]	6.28	3.57	2.83
Added moment of inertia [kg m^2]	1.05	0.11	0.06

$\lambda_5 = 0.4$. The initial relative angles of the robot were $\phi_0 = -\pi/2$[rad], $\phi_1 = 2\pi/9$[rad] and $\phi_2 = -\pi/9$[rad].

The typical experimental result is shown in Fig. 4. From Fig. 4 it can be seen that the end-tip and base follow the reference trajectories in spite of the influence of the hydrodynamic forces and the tracking errors are very small. The experimental result shows that the control performance can be improved by using the proposed method.

5 Conclusion

In this paper, for an underwater robot with vertical planar 2-link manipulator a digital RAC system was constructed. The experimental result showed the effectiveness of the proposed method.

References

[1] J. Yuh ed., *Underwater Robotic Vehicles:Design and Control*, TSI Press, 1995.

[2] T. W. McLain *et al.*, "Experiments in the Coordinated Control of an Underwater Arm/Vehicle System", *Autonomous Robots 3*, Kluwer Academic Publishers, pp. 213 – 232, 1996.

[3] G. Antonelli and S. Chiaverini, "Task-Priority Redundancy Resolution for Underwater Vehicle-Manipulator Systems", *Proc. of 1998 IEEE ICRA*, pp. 768 – 773, 1998.

[4] G. Antonelli *et al.*, "Tracking Control for Underwater Vehicle-Manipulator Systems with Velocity Estimation", *IEEE J. Oceanic Eng.*, Vol. 25, No. 3, pp. 399 – 413, 2000.

[5] N. Sarkar and T. K. Podder, "Coordinated Montion Planning and Control of Autonomous Underwater Vehicle-Manipulator Systems Subject to Drag Optimization", *IEEE J. Oceanic Eng.*, Vol. 26, No. 2, pp. 228 – 239, 2001.

[6] S. Sagara *et al.*, "Experiments of a Floating Underwater Robot with 2 Link Manipulator", *Int. J.of AROB*, Vol. 5, No. 4, pp. 215 – 219, 2001.

[7] S. Yamada and S. Sagara, "Resolved Motion Rate Control of an Underwater Robot with Vertical Planar 2-Link Manipulator", *Proc. of AROB 7th*, pp. 230 – 233, 2002.

[8] H. Maheshi *et al.*, "A Coordinated Control of an Underwater Vehicle and Robotic Manipulator", *J. Robotic Systems*, Vol. 8, No. 3, pp. 339 – 370, 1991.

[9] S. Sagara, "Digital Control of an Underwater Robot with vertical Planar 2-Link Manipulator", *Proc. of AROB 8th*, pp. 524 – 527, 2003.

[10] B. Lévesque and M. J. Richard, "Dynamic Analysis of a Manipulator in a Fluid Environment", *Int. J. Robot. Res.*, Vol. 13, No. 3, pp. 221 – 231, 1994.

[11] S. McMillan *et al.*, "Efficient Dynamic Simulation of an Underwater Vehicle with a Robotic Manipulator", *IEEE Trans. Syst., Man, Cybern.*, Vol. 25, No. 8, pp. 1194 – 1206, 1995.

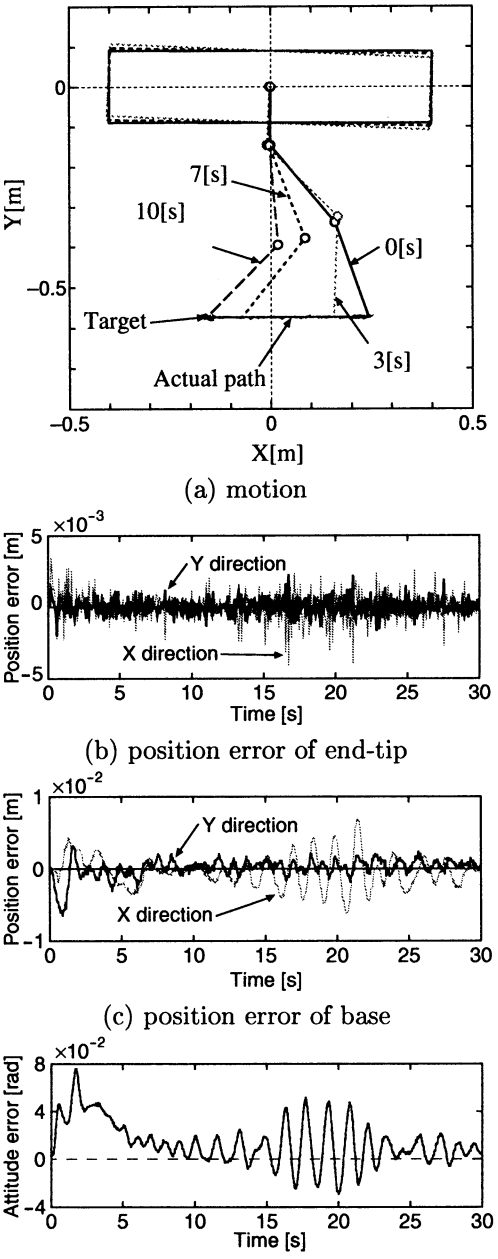


Fig. 4: Experimental result

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Fast Incremental Learning Methods Inspired by Biological Learning Behavior

Koichiro Yamauchi Takayuki Oohira Takashi Omori
Graduate School of Engineering Hokkaido University
{yamauchi, o_hira, omori}@complex.eng.hokudai.ac.jp
Keywords: Incremental Learning, Sleep Phase, Reactive Learning

Abstract

Model based learning systems such as neural networks usually forgets learned skills due to incremental learning of new instances. This is because the modification of one parameter interferes with some old memories. Therefore, the incremental learning for such learning systems needs re-learning of the old instances to avoid the forgetting.

The re-learning process, however, is a time-consuming process. In this paper, we present two types of incremental learning methods to realize quick adaptation with small quantity of resources. One approach is introducing sleep phase to gain time for the learning. The another approach introduces a 'meta-learning module' which acquires learning-skill though experiences. The system realizes 'reactive-modification' of parameters not only to memorize the new instances but also to avoid forgetting old memories using the meta-learning module.

1 Introduction

Online learning is important issue for autonomous robots, because they have to memorize new instances one by one.

Almost all of conventional online learning methods are based on the probabilistic gradient descent learning. This learning manner, however, needs a condition that the distribution of the instances should be stable. In the real world, however, the distribution of inputs are usually unstable.

To solve the problem, we usually use table look up method such as k-Nearest Neighbors (k-NNs) as the learning engine. The k-NNs not only do not forget past experiences but also memorize new instances very quickly. However, they waste huge amount of resources to record all instances. On the other hand, the model based learning systems, such as neural networks, overcome the drawback. The neural networks, however, are not suitable for the incremental learning because they forget a part of memory due to the learning of new instances. To solve the problem, the neural networks have to re-learn all instances again. However, this re-learning process is time consuming process.

To overcome the problem, we propose two types of learning systems.

2 Incremental Learning with Sleep

The first system proposed here consists of two Gaussian Radial Basis Function networks: the Fast-learning network (F-Net) and the Slow-learning network (S-Net) (Fig. 1) [1].

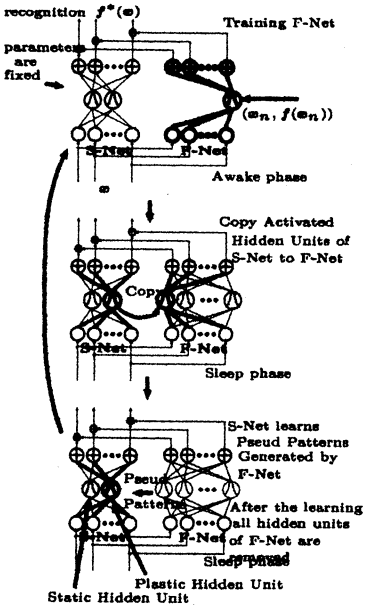


Figure 1: Structure and Behavior of the system

The system has two learning phases, the awake and the sleep phases. During the awake phase, the system recognizes instances presented, and the system output mainly comes from S-Net. If S-Net's error in the current instance exceeds a threshold, the F-Net learns the instance quickly, in the same way as k-NN [1], to compensate for the

[1] If the RBF learns each instance with one hidden unit, the

S-Net's error. The F-Net output is added to the S-Net output. As a result, the system output is always close to the desired output.

In contrast, during the sleep phase, the S-Net learns pseudo instances generated by F-Net to bring its output close to that of F-Net with a small number of hidden units. The learning method is a modified version of Minimum Resource Allocating Network (MRAN) [2], which uses a growing strategy that prunes redundant cells, as described later. The modified MRAN learns instances normally, even if the distribution of inputs is unstable. The S-Net continues learning until its outputs become close enough to those of the F-Net. After that, all hidden units of the F-Net are removed.

Strictly speaking, the S-Net learning process is applied only to some of the hidden units and corresponding weights between the hidden units and the output units. In the following text, we call the hidden units "plastic hidden units." The plastic hidden units are units activated during the adjacent awake phase. During the S-Net learning, outputs from non-plastic hidden units are fixed to zero.

Furthermore, prior to S-Net learning, parameters of the plastic hidden units are copied to the F-Net to make F-Net output agree with the desired S-Net output. Note that the S-Net memory is not interfered with by this learning, since S-Net's non-plastic hidden units do not change its parameters.

Next, the F-Net begins to learn new instances again. However, S-Net needs a long learning period, which is much longer than that of F-Net, to complete the learning, meaning that the reduced speed of redundant hidden units is much slower than that of the increased speed of F-Net's hidden units. Therefore, if the system learns both S- and F-Net simultaneously, the number of hidden units still continue to increase until the system approximates the total area of input space. To solve this problem, F-Net learning is arrested during S-Net learning. Therefore, the system stops reading new novel instances during the learning period, resulting in alternating learning processes for S- and F-Net.

Although the system has to stop reading new instances during periodic intervals, just as humans stop reading during sleep, the system perfectly memorizes new instances and reduces redundant hidden units without forgetting past memory.

In the following text, $\boldsymbol{f}_f^*(\boldsymbol{x}, \boldsymbol{\theta}_f)$ and $\boldsymbol{f}_s^*(\boldsymbol{x}, \boldsymbol{\theta}_s)$ denote the output vectors of F-Net and S-Net, respectively. Here, $\boldsymbol{\theta}_*$ and $\boldsymbol{x}$ denote the parameter vector of the networks and input vector, respectively. The network's output is the same as that of RBF. For example, the i-th output of F-Net is

$$f_{fi}^*(\boldsymbol{x}, \boldsymbol{\theta}_f) = \sum_{\alpha} c_{i\alpha}^f \phi_{F_\alpha}(\boldsymbol{x}), \tag{1}$$

$$\phi_{F_\alpha}(\boldsymbol{x}) \equiv \exp\left(-\frac{\|\boldsymbol{x}_j - \boldsymbol{u}_{F\alpha}\|^2}{2\sigma_{F\alpha}^2}\right), \tag{2}$$

where $c_{iF\alpha}^f$ is the connection strength between the $F\alpha$-th hidden unit and the i-th output cell. Here, $\phi_{F_\alpha}(\boldsymbol{x})$ denotes the output value from the $F\alpha$-th hidden unit, while $\boldsymbol{t}_{F\alpha}$ and $\sigma_{F\alpha}$ are the reference vector and the deviation, respectively.

2.1 Awake Phase

During the awake phase, the system recognizes known inputs by yielding the sum of the outputs from the F-Net and S-Net while the F-Net learns unknown instances quickly. Let $\boldsymbol{f}^*(\boldsymbol{x})$ and $\boldsymbol{D}(\boldsymbol{x})$ be the final output of the system and the desired output to $\boldsymbol{x}$, respectively, where $\boldsymbol{f}^*(\boldsymbol{x}) = \boldsymbol{f}_f^*(\boldsymbol{x}, \boldsymbol{\theta}_f) + \boldsymbol{f}_S^*(\boldsymbol{x}, \boldsymbol{\theta}_S)$.

The F-Net learns a new unknown instance $(\boldsymbol{x}_n, \boldsymbol{D}(\boldsymbol{x}_n))$ as the pseudo code show in Fig 2. In the figure, ε denotes a learning rate.

Note that the F-Net also allocates new basis functions in the same way as the resource-allocating network (RAN) [3]. The allocation procedure is similar to that of RAN; however, the allocation condition is not only on the distribution of the basis function of F-Net, but also on that of S-Net.

foreach $\boldsymbol{x}_n$
if $\min_{S\alpha} \|\boldsymbol{x}_n - \boldsymbol{u}_{S\alpha}\| > \epsilon_D$ and
$\min_{F\alpha} \|\boldsymbol{x}_n - \boldsymbol{u}_{F\alpha}\| > \epsilon_D$ and
$\|\boldsymbol{D}(\boldsymbol{x}_n) - \boldsymbol{f}^*(\boldsymbol{x}_n)\|^2 > \epsilon_E$ **then**
allocate new cell $F\alpha$
$\boldsymbol{u}_{F\alpha} = \boldsymbol{x}_n$
$\boldsymbol{c}_{F\alpha} = \boldsymbol{D}(\boldsymbol{x}_n) - \boldsymbol{f}^*(\boldsymbol{x}_n)$
$\sigma_{F\alpha} = \min(\lambda_F \min_{S\alpha} \|\boldsymbol{x}_n - \boldsymbol{u}_{S\alpha}\|, \lambda_F \min_{F\beta \neq F\alpha} \|\boldsymbol{x}_n - \boldsymbol{u}_{F\beta}\|, \sigma_{\max})$
else
$\boldsymbol{u}_{F\alpha} := \boldsymbol{u}_{F\alpha} - \varepsilon \nabla_{\boldsymbol{u}_{F\alpha}} \|\boldsymbol{D}(\boldsymbol{x}_n) - \boldsymbol{f}^*(\boldsymbol{x}_n)\|^2$
$\boldsymbol{c}_{F\alpha} := \boldsymbol{c}_{F\alpha} - \varepsilon \nabla_{\boldsymbol{c}_{F\alpha}} \|\boldsymbol{D}(\boldsymbol{x}_n) - \boldsymbol{f}^*(\boldsymbol{x}_n)\|^2$
$\sigma_{F\alpha} := \sigma_{F\alpha} - \varepsilon \nabla_{\sigma_{F\alpha}} \|\boldsymbol{D}(\boldsymbol{x}_n) - \boldsymbol{f}^*(\boldsymbol{x}_n)\|^2$

Figure 2: Learning Procedure of F-Net

2.2 Sleep Phase

During the sleep phase, the S-Net learns the outputs from the F-Net and prunes redundant hidden units.

RBF shows k-NN-like behavior.

However, the F-Net output is not the desired output for S-Net but the error of S-Net. Therefore, the system converts the output to the desired output by copying some of the hidden units in the S-Net to F-Net. The copied hidden units are the "plastic hidden units" that were activated during the adjacent awake phase. [2]. As a result, the output becomes the desired output for S-Net, which reflects not only the new instances, but also the old memories that will be re-learned to avoid forgetting. Note that the outputs of the non-plastic hidden units are fixed to zero during this learning to ensure consistency.

Next, S-Net begins to learn the pseudo patterns $(\hat{\boldsymbol{x}}_F, \boldsymbol{f}_F^*(\hat{\boldsymbol{x}}_F))$ generated by the F-Net, where $\hat{\boldsymbol{x}}_F$ is the pseudo input vector that is generated by adding random noise to the centroid $\boldsymbol{u}_{F_j}$ of each hidden unit.

The system randomly chooses one hidden unit in the F-Net every time one pseudo input is generated. If the system chooses the j-th hidden unit, the pseudo input vector $\hat{\boldsymbol{x}}_F$ is represented as follows;

$$\hat{\boldsymbol{x}}_F = \boldsymbol{u}_{F_j} + \sigma_{F_{max_j}} \boldsymbol{\kappa}, \tag{3}$$

where $\boldsymbol{\kappa}$ is a random unit vector. Here, $\sigma_{F\max_j}$ denotes the maximum distance between the input pattern and the centroid of the j-th hidden unit, and $\sigma_{F\max_j}$ is determined during the awake phase.

The learning method for S-Net is a modified version of the Minimum Resource Allocating Network(MRAN) learning algorithm [2], which is an online learning method with a pruning strategy. In the pruning strategy, S-Net prunes any hidden unit whose contribution ratio is smaller than a threshold. The contribution ratio of the S_j-th hidden unit is the relative magnitude of outputs, as per the below equation.

$$r_j(\hat{\boldsymbol{x}}_{F_k}) = \|\boldsymbol{c}_j \phi_{S_j}(\hat{\boldsymbol{x}}_{F_k})\| / \max_i \|\boldsymbol{c}_i \phi_{S_i}(\hat{\boldsymbol{x}}_{F_k})\|, \tag{4}$$

where $\boldsymbol{c}_j$ is the connection strength vector between the S_j-th hidden unit and the output units.

The pruning algorithm is given below.

for each pseudo instance $(\hat{\boldsymbol{x}}_F, \boldsymbol{f}_F^*(\hat{\boldsymbol{x}}_F))$
  for each S_j-th hidden unit
  if $\|\hat{\boldsymbol{x}} - \boldsymbol{u}_{S\alpha}\| < \sigma_{S\alpha}$ then
    Calculate the contributing ratio $r_{S_j}(\hat{\boldsymbol{x}}_F)$ with Eq(4)
    if $r_{S_j}(\hat{\boldsymbol{x}}_F) < \delta$ for M consecutive $\hat{\boldsymbol{x}}_F$ then
      prune the S_j-th hidden unit;

[2]During awake phase, each hidden unit in S-Net checks its "Activated-flag" when the unit is activated. If $\phi_{F\alpha}(\boldsymbol{x})$ exceeds the threshold ϵ_o, then $ActivatedFlag_i = true$. If $ActivatedFlag_i = true$, then the i-th hidden unit is copied to the F-Net.

During the awake phase, if the j-th hidden unit center is the closest to the current input and $\sigma_{F\max\ j} < \|\boldsymbol{u}_{Fj} - \boldsymbol{x}_n\|$, then $\sigma_{F\max\ j}$ is updated to be the current distance: $\sigma_{F\max\ j} := \|\boldsymbol{u}_{Fj} - \boldsymbol{x}_n\|$.

for $n = 1$ to $N_{optimize}$ **times**
  Get pseudo pattern $\hat{\boldsymbol{x}}_F$ from the F-Net.
  $e_{rmsn} := \sqrt{\frac{\sum_{i=t-M}^{T} \|\boldsymbol{f}_F^*(\hat{\boldsymbol{x}}_{Fi}) - \boldsymbol{f}_S^*(\hat{\boldsymbol{x}}_{Fi})\|^2}{M}}$
  $e_n = \max\{\epsilon_{max}\gamma^n, \epsilon_{min}\}$, where $\gamma < 1$ is a decay constant
  if $\min_{S\alpha} \|\boldsymbol{x}_F - \boldsymbol{u}_{S\alpha}\| > e_n$ and
  $\|\boldsymbol{f}_F^*(\hat{\boldsymbol{x}}_F) - \boldsymbol{f}_S^*(\hat{\boldsymbol{x}}_F)\| > e_E$ and
  $e_{rmsn} > e_{rmsn}$ and
  Number of cells is less than that of F-Net
  then
    allocate new cell $S\alpha$
    $\boldsymbol{u}_{S\alpha} = \hat{\boldsymbol{x}}_F$
    $\boldsymbol{c}_{S\alpha} = \boldsymbol{f}_F^*(\hat{\boldsymbol{x}}_F) - \boldsymbol{f}_S^*(\hat{\boldsymbol{x}}_F)$
    $\sigma_{S\alpha} = \min(\lambda_S \min_{S\beta} \|\hat{\boldsymbol{x}}_F - \boldsymbol{u}_{S\beta}\|, \sigma_{\max})$
  else
    $\boldsymbol{u}_{S\alpha} := \boldsymbol{u}_{S\alpha} - \varepsilon \nabla_{\boldsymbol{u}_{S\alpha}} \|\boldsymbol{f}(\hat{\boldsymbol{x}}_F) - \boldsymbol{f}^*(\hat{\boldsymbol{x}}_F)\|^2$
    $\boldsymbol{c}_{S\alpha} := \boldsymbol{c}_{S\alpha} - \varepsilon \nabla_{\boldsymbol{c}_{S\alpha}} \|\boldsymbol{f}(\hat{\boldsymbol{x}}_F) - \boldsymbol{f}^*(\hat{\boldsymbol{x}}_F)\|^2$
    $\sigma_{S\alpha} := \sigma_{S\alpha} - \varepsilon \nabla_{\sigma_{S\alpha}} \|\boldsymbol{f}(\hat{\boldsymbol{x}}_F) - \boldsymbol{f}^*(\hat{\boldsymbol{x}}_F)\|^2$
  endif
  Execute the pruning algorithm(see text in detail)
endfor

Figure 3: Learning procedure of S-Net

Fig. 3 shows the whole learning procedure of S-Net. In this figure, e_{rmsn} denotes the mean square error of S-Net.

The leaning process is repeated $N_{optimize}$ times, where $N_{optimize}$ is set N_o times the number of hidden units of F-Net. In the experiment described later, N_o was set to 200. After learning by S-Net, the system removes all hidden units of F-Net, after which the awake phase is restarted.

3 Reactive Incremental Learning

The incremental learning system described above reduces apparent learning time by introducing sleep phase. On the opposite, it is well known that we usually understand new situation quickly when we encounter it. Therefore, there are possibility that we use a kind of reactive learning system to realize quick understanding or learning of the new situation. From this point of view, we proposed a novel incremental learning system which has an Meta-Learning network [4].

This system consists of three parts, a pattern buffer, and main- and meta-learning modules. The main-learning module approximates a function between inputs and outputs of instances. This module normally modifies its parameters using gradient descent learning of all instances stored in the pattern buffer. The pattern buffer stores a part of old samples. In contrast, the meta-learning module tries to predict appropriate change in parameters of main-learning module according to past experiences of incremental learning. The meta-learning module tries to predict an appropriate change in parameter which makes the output of the main-learning module fit to the new instance with avoiding the loss of previously learned skills. If the meta-learning module is confident about the current prediction, the main-learning module modifies its parameters according to the predicted change and not the gradient descent learning. Thus, the system sometimes finishes the incremental learning with just one reactive-modification of the parameters based on the predicted change in parameters. In fact, however, the system usually needs additional gradient descent learning with a small number of iterations to reduce the least error.

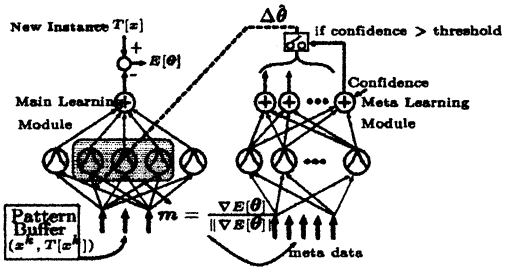


Figure 4: Structure of the system

The whole learning procedure is summarized as follows.

(1) Store parameter of the main learning network to θ^{old}

(2) Present a new instance $(x, T[x])$ and calculate error $E[\theta]$

(3) Get meta data m (see 3.1) from the current error $E[\theta]$ of the main module.

(4) Calculate output of the meta-learning module.

(5) Modify parameters of the main module using the outputs of step (4)

(6) Push $(x, T[x])$ into the pattern buffer.

(7) Learning of instances stored in the buffer with a gradient descent method.

(8) Have the meta-learning network learn $\theta_m^{new} - \theta_m^{old}$. (see 3.1)

3.1 Meta-Learning Module

The meta-learning module predicts appropriate parameter changes for the main-learning modules. Note that the main- and meta-learning modules are the RBF described by Eq(1)(2). Especially, the meta-learning module is the resource allocating network [3]. The input vector for the meta-learning module is basically a normalized gradient vector of the main network's least square error of the new instance. However, if the module observes entire dimension of the gradient vector, the module cannot find a rule behind the data because of large-scale variations.

To solve the problem, the system extracts a subdimension from the gradient vector: the gradient vector of the three hidden units with centroids closest to current input x. Let θ_m be the parameters of the three hidden units:

$$\theta_m \equiv (c_{i1}, \sigma_{i1}, u_{i1}^T, c_{i2}, \sigma_{i2}, u_{i2}^T, c_{i3}, \sigma_{i3}, u_{i3}^T)^T \quad (5)$$

where $i1$,$i2$ and $i3$ denote the indexes of the three hidden units, and 'T' denotes the transportation. $i1$,$i2$ and $i3$ are arranged in the ascending order of the distance between respective centroid and the current input. We chose the meta-data to be the normalized gradient vector of the main-learning module of the three hidden units $m = D/\|D\|$ where

$$D \equiv \nabla_{\theta_m} E[\theta] \mid_{\theta_m = \theta_m^{old}} \quad (6)$$

$E[\theta]$ denotes the current error of the main-learning module: $E[\theta] = \|T[x] - f^*(x, \theta)\|^2$.

The met-learning module learns the actual change in parameters of the main-learning module. The changes in parameters are calculated as follows. First of all, the predicted changes in parameters (the output of the meta-learning module) are added to the old parameters.

$$\theta_m' = \theta_m + \|D\| g^*(m, \xi) \quad (7)$$

Note that $\|D\|$ is multiplied to the output vector of the meta-learning network to expand scale of the output vector to an appropriate length. The reason for this adjustment is that the meta-learning network always learns using a normalized vector to avoid the affects of variation in vector length.

Then, the parameter vector of the main-learning module becomes θ', which is a refrection of θ_m'. The main-learning network learns the instances stored in the pattern buffer until the total error of the main-

learning module is less than threshold Err_{min}.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla \left\{ \sum_{k \in buffer} \|\boldsymbol{T}(\boldsymbol{x}^k) - \boldsymbol{T}^*(\boldsymbol{x}^k, \boldsymbol{\theta}_t)\|^2 \right\} \tag{8}$$

where η (> 0) is the speed of the learning and $\boldsymbol{\theta}_0 = \boldsymbol{\theta}'$.

Then, the parameter vector becomes $\boldsymbol{\theta}^{new}$. The meta-learning module learns the actual change in parameters of the three hidden units: $\boldsymbol{\theta}_m^{new} - \boldsymbol{\theta}_m^{old}$, where $\boldsymbol{\theta}_m^{old}$ denotes the initial parameter vector. Therefore,

$$\boldsymbol{\xi} := \boldsymbol{\xi} - \eta \nabla_{\boldsymbol{\xi}} \left\{ \|\Delta\boldsymbol{\theta}_*/\|\boldsymbol{D}\| - \boldsymbol{g}^*(\boldsymbol{m}, \boldsymbol{\xi})\|^2 + Err_c(\boldsymbol{m}, \boldsymbol{\xi}) \right\} \tag{9}$$

where $\boldsymbol{\xi}$ denotes the parameter vector of the meta-learning network and $\Delta\boldsymbol{\theta}_* \equiv \boldsymbol{\theta}_m^{new} - \boldsymbol{\theta}_m^{old}$. In Eq(9), $Err_c(\boldsymbol{m}, \boldsymbol{\xi})$ denotes the error of the additional unit (see Figure 4), which outputs the 'magnitude of confidence' for the current prediction. The desired output for this additional unit should be large when the current prediction is correct and small when incorrect. So, we defined the desired output of this additional unit T_c as

$$T_c \equiv \exp\left(-\|\Delta\boldsymbol{\theta}_*/\|\boldsymbol{D}\| - \boldsymbol{g}^*(\boldsymbol{m}, \boldsymbol{\xi})\|^2\right) \tag{10}$$

Then, $Err_c(\boldsymbol{m}, \boldsymbol{\xi}) = (T_c - C(\boldsymbol{m}, \boldsymbol{\xi}))^2$ where $C(\boldsymbol{m}, \boldsymbol{\xi})$ denotes the output value of the additional unit. Eq(9) is executed only once every after the learning cycle of the main-learning module.

4 Experiments

This section shows the examples of behaviors of the two incremental learning systems.

4.1 Incremental Learning with Sleep

The parameters used in this system were predetermined empirically though preliminary simulation studies. The parameters used are listed in Table 1.

Figures 5 shows the responses of the Incremental Learning system with Sleep (ILS), 3- and 5-Nearest Neighbors to the Servo dataset, which has 167 samples. In this test, the systems have to predict continuous class from 12 input values. The ILS learns 20 new samples in each awake phase and repeates the awake and sleep phases 9 times to complete the learning of the 167 samples. The test was repeated 50 times and the results are averaged over the 50 trials. In the figure, the mean square error of the system to all instances in the servo datasets are examined immediately after each awake and sleep phases. Furthermore,

parameter	explanation	value
ε	learning speed	0.001
ϵ_E	Allocation threshold for F-Net	0.0001
ϵ_D	Allocation threshold for F-Net	0.01
λ_F	Overlap factor for F-Net	0.3
λ_S	Overlap factor for S-Net	1
ϵ_{max}	Allocation threshold of S-Net	1
ϵ_{min}	Allocation threshold of S-Net	0.01
e'_{rmse}	Allocation threshold of S-Net	$0.01 \times N_y$
δ	Pruning threshold	0.3
M	Consecutive Observations	10
T	Buffer size for calculating MSE	100
N_o	Number of pseudo instance per cell	200
ϵ_o	for checking whether each hidden unit is activated or not.	0.00001

Table 1: Parameters used (N_y denotes number of output dimension)

the quantitiy of resources immediately after each sleep phase and the maximum quantity of resources needed during the awake and sleep phases are also examined. It is clear that the quantity of resources of the ILS is increased during the learning, but is reduced after the sleep phase. Moreover, the mean square error immediately after each sleep phase is also less than that of each awake phase. This means that the generalization ability of the system is improved by pruning redundant hidden units during the sleep phase.

4.2 Reactive Incremental Learning

The pre-determined parameters for the meta-learning module were $\varepsilon_a = 0.7$ (allocation threshold), $\eta = 0.1$ (learning rate) and $\gamma = 0.65$ (the overlap factor). The size of pattern buffer was set to 20. The initial standard deviation of the hidden units of the main-learning module was set to 0.3. The gradient descent learning of the main-learning module (step (7) in 3) was continued until least square error to all instances stored in the pattern buffer was less than 0.00015. The confidence threshold was set to 0.995. In the preliminary simulation study, we empirically found these values of the parameters make the system work appropriately.

The dataset used is the Mackey Glass Chaos of UCI

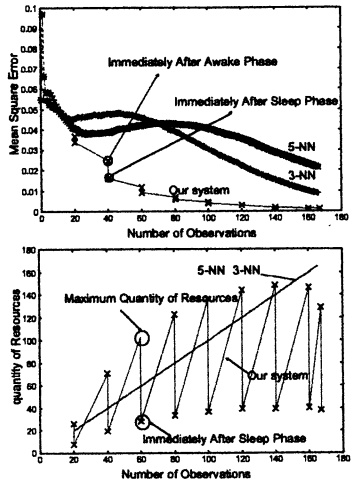


Figure 5: Mean square error and quantity of resources in Servo dataset

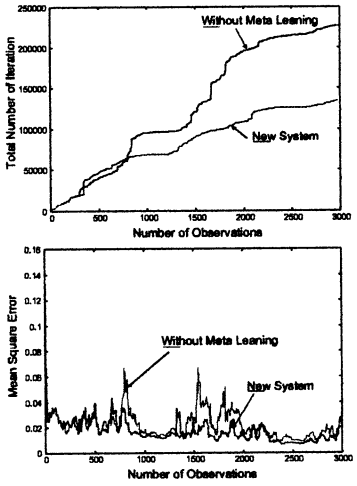


Figure 6: Total Number of Iterations and Mean Square Error for the Mackey Glass Chaos.

Machine Repository, which is a chaos time series.

This dataset consists of 3000 sets of the four dimensional input pattern and the corresponding desired output value $T[\boldsymbol{x}]$. The centroids of the main-learning module were initially positioned at lattice points in the cubic area: x_i in the interval $[0.5, 1.2]$ for $i = 1, 2, 3, 4$. The total number of the centroids was 81.

Figure 6 shows the total number of iterations and the mean square error of our new system and the system without the meta-learning module. We also see that the number of iterations of our new system decreases greatly compared with that of the system without meta-learning module.

5 Conclusion and Discussion

We showed two incremental learning systems, which are inspired by biological learning behavior. The first system reduces apparent learning time by introducing sleep phase. This system realizes almost all of the incremental learning process during sleep phase. Instead, the learning during awake phase is very simple and quick learning. Therefore, the first system reduces apparent learning time. The experimental results showed that the system realizes not only quick learning during awake phase but also improving generalization ability by reducing redundant hidden units during sleep phase.

On the other hand, the second system realizes actual reduction of learning time by introducing reactive-learning using the meta-learning network. The experimental result showed that this system works well in chaos time series, in which appears similar patterns repeatedly. The performance of the system, however, is depends on the order of presentation of instances. Therefore, the second system works well if there are some hidden laws in the presentation of the patterns.

Our future work is improving the abilities of the two systems and combine the two methods to realize really quick incremental learning.

References

[1] Koichiro Yamauchi. Incremental learning with sleep –application for predicting posterior distribution–. In *International Joint Conference on Neural Networks, IJCNN2003*, volume 4, pages 2776–2781, July 2003.

[2] Lu Yingwei, N. Sundararajan, and P. Saratchandran. A sequential learning scheme for function approximation using minimal radial basis function neural networks. *Neural Computation*, 9:461–478, 1997.

[3] John Platt. A resource allocating network for function interpolation. *Neural Computation*, 3(2):213–225, 1991.

[4] Takayuki Oohira, Koichiro Yamauchi, and Takashi Omori. Meta-learning for fast incremental learning. In Okayay Kaynak, Ethem Alpaydin, Erkki Oja, and Lei Xu, editors, *ICANN/ICONIP 2003 LNCS2714*, pages 157–164. Springer-Verlag, June 2003.

Proc. of The Ninth Int. Symp. on Artificial Life and Robotics (AROB 9th '04)
Beppu, Oita, Japan, 28-30 January, 2004

Self-Organizing Neural Networks for Concept Formation

Noriyasu Homma
Faculty of Medicine
Tohoku University
Sendai, 980-8575

Masao Sakai
Information Synergy Center
Tohoku University
Sendai, 980-8576

Kenichi Abe
Graduate School of Engineering
Tohoku University
Sendai, 980-8579

Abstract

We propose a self-organizing neural structure for forming a feature space representation of concepts. An essential core of this self-organization is an appropriate combination of an unsupervised learning with incomplete information for the dynamic changing and an extended Hebbian rule for a signal-driven spatial changing. A concept formation problem requires the neural network to acquire finally the *complete* feature space structure of concept information using an *incomplete* observation of the concept. The informational structure can be stored as the connection structure of self-organizing network by using the two rules: The Hebbian rule can create a necessary connection, while unsupervised learning can delete unnecessary connections. Finally concept formation ability of the proposed neural network is proven under some conditions.

1 Introduction

In neural networks, synaptic weights are considered to store knowledge of the past experiences [1]. Thus, learning and adaptation algorithms provide how to change the weights in order to accomplish given missions. Although the supervised backpropagation learning is one of the most successful neural applications, unsupervised learning or self-organization is an attractive function in biological neural networks. For example, Hebbian learning is a biologically based unsupervised learning [2] and the self-organizing map (SOM) can learn a topological relation of the environment as in its network structure [3].

On the other hand, cognition using incomplete information or observation is an another attractive and intrinsic function in brain. The SOM, however, needs the complete observation of the input information. This implies the SOM cannot self-organize its connection weights by using incomplete information of the inputs.

In this paper we propose a self-organizing neural structure for storing feature space representation of logical concepts from incomplete observation by using a neuron model with dynamic and spatial changing weights [4]. To form the complete informational structure of concepts, (i) a necessary connection structure is created by an extended Hebbian rule and (ii) unnecessary connections are deleted by an unsupervised competitive learning. An ability of the proposed neural network for acquiring the informational structure is discussed by using a concept formation problem.

2 Concept Formation Problem

We define a vector representation of a *concept*, $\boldsymbol{x}$, using a set of feature variables $\{x_1, x_2, \ldots, x_n\}$

$$\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T \in \Re^n \tag{1}$$

Let us consider a set of N vector representations of N concepts, S, given as

$$S = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^N\} \tag{2}$$

where $\boldsymbol{x}^i$, $i = 1, 2, \ldots, N$, are the vector representations of concepts i. A neuron *receives* the complete information of concept i, $\boldsymbol{x}^i$, but can *recognize* only an incomplete observation, $\hat{\boldsymbol{x}}^i = [\hat{x}_1^i, \hat{x}_2^i, \ldots, \hat{x}_n^i]^T$, through its synaptic connection vector $\boldsymbol{w} = [w_1, w_2, \ldots, w_n]^T$

$$\hat{x}_j^i = w_j x_j^i, \quad j = 1, 2, \ldots, n \tag{3}$$

Let the initial weight vector be $\boldsymbol{w}(0) = 0$ and it will be developed to a connection structure $\boldsymbol{w} \neq 0$. For example, at the initial stage a neuron cannot *recognize* any input information through the initial connections, $\boldsymbol{w}(0) = 0$.

Here the goal of the ith concept formation by the ith neuron is to satisfy the following condition

$$\hat{\boldsymbol{x}}^i \longrightarrow \boldsymbol{x}^i \tag{4}$$

In other words, a concept formation defined in this paper is an acquisition process of *awareness* structure for observing the complete information by developing connection weights $\boldsymbol{w}^i$. In this sense, the connection weights imply a strength of *awareness* for the received information.

To seek a meaningful solution [5], let us use the unipolar binary representation for the feature values, that is,

$$x_i \in \{0,1\}, \quad i = 1,2,\ldots,n \tag{5}$$

then $x_i^2 = x_i$. This implies that the following weights vector is also the solution of Eq. (4)

$$\boldsymbol{w}^i \longrightarrow \boldsymbol{x}^i \tag{6}$$

In this case, the neural structure will converge the *informational structure* of the concept.

Finally, formation of a set of concepts S using a neural network with M neurons, $(M \geq N)$, can be represented by

$$S \subset S_{NN} = \{\boldsymbol{w}^1, \boldsymbol{w}^2, \ldots, \boldsymbol{w}^M\} \tag{7}$$

In addition to this definition of the goal, human cognition may be subjective and relative. Thus even if an incomplete information $\hat{\boldsymbol{x}}^i$ is observed through underdeveloped connections, it can be recognized as *concept* i at current deepness of understanding ("mass" is recognized as "mass" even if we do not know about the *mass–energy equivalence* "$E = mc^2$" discovered by A. Einstein.). We assume that the neural cognition is a process of matching its inner awareness structure $\boldsymbol{w}$ with an observed informational structure $\hat{\boldsymbol{x}}$, not with the complete informational structure $\boldsymbol{x}$. The cognition *result* may then be independent of the deepness of understanding, although the deepness is improved as new attributes of the concept are discovered.

For simplicity let $0 \leq w_i \leq 1$, $i = 1,2,\ldots,n$ then the deepness of understanding can be represented by the total amount of awareness given as the sum of the weights $W = \sum_{i=1}^{n} w_i$. Thus the neural output as a result of the cognition process for concept i can be defined as

$$y = \begin{cases} \dfrac{1}{W}\displaystyle\sum_{j=1}^{n} \hat{x}_j^i = \frac{1}{W}\boldsymbol{w}^T\boldsymbol{x}^i & (W \neq 0) \\ 0 & (W = 0) \end{cases} \tag{8}$$

The normalized output implies the ratio of a sum of the observed information to the total amount of awareness. If a neuron is aware of a subset of the complete informational structure, the output is equal to 1, otherwise the output is less than 1, regardless of the total amount of awareness W.

3 Self-Organizing Neural Structure with Dynamic and Spatial Changing Weights

3.1 General Model of Discrete-Time Neurons with Dynamic and Spatial Changing Weights

Fig. 1 shows the model of a discrete-time neuron with dynamic and spatial changing weights (DSCWs) [4]. In this neural model, the following variables are defined: $\boldsymbol{x}(k) = [x_1(k), x_2(k), \ldots, x_n(k)]^T \in \Re^n$: neural input vector; $\boldsymbol{r}(k) = [r_1(k), r_2(k), \ldots, r_n(k)]^T \in \Re^n$: spatial distance vector between the sensory devices or another neuron's axon branches and the corresponding target dendrites (Fig. 1); $\boldsymbol{w}(\boldsymbol{r}(k)) = [w_1(r_1(k)), w_2(r_2(k)), \ldots, w_n(r_n(k))]^T \in \Re^n$: weight vector as a function of the spatial distance vector $\boldsymbol{r}$; and $y(k) \in \Re$: neural output. The input $\boldsymbol{x}(k)$ to the neuron may be external signals or outputs from other neurons, but no self-feedback connection is considered in this paper.

By the definition of the weight vector, the derivatives dw_i/dr_j are 0 for $i \neq j$. Then the spatial changes in the weights are defined by the diagonal matrix as

$$\frac{d\boldsymbol{w}(\boldsymbol{r})}{d\boldsymbol{r}} = \text{diag}\left(\frac{dw_1}{dr_1}, \frac{dw_2}{dr_2}, \cdots, \frac{dw_n}{dr_n}\right) \tag{9}$$

In general a necessary condition to form the ith synaptic weight is that the spatial distance r_i is sufficiently short. For example, the following function satisfies this condition:

$$w_i(r_i) = \begin{cases} w_0, & (r_i < r_0) \\ 0, & (r_i > r_0) \end{cases} \tag{10}$$

where r_0 is a constant or so-called threshold.

A discrete-time *dynamic change* in the connecting weight is defined as

$$\boldsymbol{w}(\boldsymbol{r}(k+1)) = \boldsymbol{w}(\boldsymbol{r}(k)) + \Delta\boldsymbol{w}(\boldsymbol{w}(\boldsymbol{r}(k)), \boldsymbol{x}(k), y(k)) \tag{11}$$

where $\Delta\boldsymbol{w} = [\Delta w_1, \Delta w_2, \ldots, \Delta w_n]^T$ is a vector function describing discrete-time changes in the weight vector. Note that this discrete-time dynamic changes can be used for the discrete-time *learning* scheme that provides how to change the *existing* synaptic weights.

To create new synaptic connections, another mechanism such as *structural adaptation* is needed. A dynamic change in the distance is introduced as a structural adaptation mechanism given by

$$\boldsymbol{r}(k+1) = \boldsymbol{r}(k) + \Delta\boldsymbol{r}(\boldsymbol{r}(k), \boldsymbol{x}(k), y(k)) \tag{12}$$

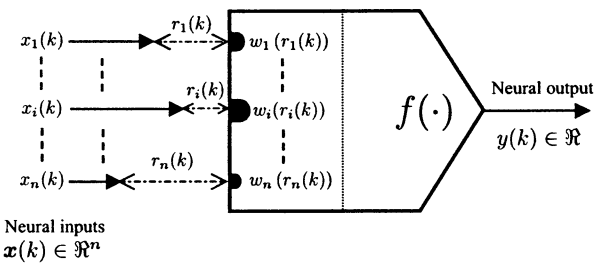


Figure 1: A discrete-time neuron with dynamic and spatial changing weights (DSCWs).

where $\Delta \boldsymbol{r} = [\Delta r_1, \Delta r_2, \ldots, \Delta r_n]^T$ is a vector function describing discrete-time changes in the distance vector. Note that both the dynamic changes in Eqs. (11) and (12) imply the changes in the distance $\boldsymbol{r}(k)$. Eq. (11) implies a *learning* algorithm for only *existing* weights $w_i \neq 0$, that is, it only *changes* a strength of awareness. On the other hand, *discovering* a new attribute, that is a creation of a new synaptic connection, is not a subset of the *change*, but achieved by a structural adaptation given in Eq. (12). A specific combination of these two rules can be used for a specific task.

Substituting the dynamic and spatial changing weights into Eq. (8) for $W \neq 0$, the neural output, as a result of neural cognition of a concept, is defined as

$$\begin{aligned} y(k) &= f(\boldsymbol{x}(k), \boldsymbol{w}(\boldsymbol{r}(k))) \in \Re \\ &\equiv \frac{1}{W}(\boldsymbol{w}(\boldsymbol{r}(k))^T \boldsymbol{x}(k) \end{aligned} \tag{13}$$

where f is an activation function (Fig. 1).

3.2 Unsupervised Competitive Learning

We select a neuron c whose output is the largest among the neural network

$$c = arg \max_j y_j(k) \tag{14}$$

If there are some neurons c_j, $j = 1, 2, \ldots$, that satisfy Eq. (14), then the neuron with the largest sum of weights is selected.

$$c = arg \max_{c_j} W^{c_j}(k) \tag{15}$$

If there are still several candidates, a neuron that satisfy Eq. (15) is selected randomly. Then the neural outputs are re-defined as $y_j(k) = 0$, $(j \neq c)$, and $y_c(k) = y_c(k)$.

The learning rule in Eq. (11) is achieved by using observed information $\hat{\boldsymbol{x}}$ instead of the complete information $\boldsymbol{x}$

$$\begin{cases} w_j^c(k+1) &= w_j^c(k) + \gamma[\hat{x}_j(k) - w_j^c(k)] \\ &= w_j^c(k)[1 + \gamma(x_j(k) - 1)] \\ w_j^i(k+1) &= w_j^i(k), \quad (i \neq c) \end{cases} \tag{16}$$

where $\gamma > 0$ is a learning constant. For simplicity, let $\gamma = 1$ then this competitive learning rule can be written as

$$\begin{cases} w_j^c(k+1) &= w_j^c(k) x_j(k) = \hat{x}_j(k) \\ w_j^i(k+1) &= w_j^i(k), \quad (i \neq c) \end{cases} \tag{17}$$

In the case of $w_j^c(k) = 1$, note that if $\hat{x}_j(k) = 1$, that is $x_j(k) = 1$, then this feature is needed to be recognized and thus the connection structure will not be changed $w_j^c(k+1) = 1$. On the other hand, if $\hat{x}_j(k) = 0$, that is $x_j(k) = 0$, this feature is an unnecessary information and thus this connection should be disappear $w_j^c(k+1) = 0$. Thus this learning rule can delete unnecessary connections. However, if $w_j^c(k) = 0$, this learning rule cannot change this weight. The creation of a new connection can be achieved by the following structural adaptation.

3.3 A Hebbian Rule for Structural Adaptation

Using the spatial changing function in Eq. (10), if a distance is sufficiently small the synaptic weight is relatively large. neuron j, w_{ji}, can be formed by growth of an axon branch of neuron i and a dendrite of neuron j. In this paper Eq. (12) is achieved by an extended Hebbian rule as follows:

$$\begin{aligned} r_{ji}(k+1) &= r_{ji}(k) - \eta r_{ji}(k) x_i(k) y_j(k) \\ &= r_{ji}(k)(1 - \eta x_i(k) y_j(k)) \end{aligned} \tag{18}$$

where $\eta > 0$ is a growing constant and x_i and y_j are outputs of sensory and cognitive neurons, respectively. Here, if the two neurons i and j fire simultaneously, the distance $r_{ji}(k+1) = r_{ji}(k)(1-\eta)$. On the other hand, they do not fire simultaneously, the distance will not change. For simplicity, let $\eta = 1$, $x_i \in \{0, 1\}$, $y_c = 1$, and $y_{j \neq c} = 0$ then

$$r_{ji}(k+1) = \begin{cases} 0, & (x_i(k) y_c(k) = 1) \\ r_{ji}(k), & (x_i(k) y_j(k) = 0) \end{cases} \tag{19}$$

The input information x_i, however, can be recognized only through the *existing* synaptic connection with $w_i^c = 1$ since if $w_i^c = 0$ then the recognizable input $\hat{x}$ is always equal to 0, $\hat{x}_i = w_i^c x_i = 0$, regardless of the input value of x_i. To avoid this dilemma, let us introduce the following supplemental growing scheme: If the ith sensory neuron fires, $x_i = 1$, distances between its axon and the dendrites of some possible target neurons, r_{ji}, become sufficiently short to form synaptic connections *temporary*. Since the ith sensory neuron does not know whether a target cognitive neuron is the candidate neuron c or not, target neurons with $w_i^j = 0$ are selected randomly in this paper. Then according to Eq. (19), the temporal synaptic connections will be *deleted* for the target neurons that do not fire simultaneously, $y_{j \neq c} = 0$, but if the target is the candidate neuron c, the connection will be formed permanently since these two neurons i and c fire simultaneously, $x_i y_c = 1$. Note that, thus only a necessary connection structure such that $x_i(k) y_c(k) = 1$ can be created by using Eq. (19).

Such random search of formation and deletion needs so-called *trial and error* mechanism. In general if the number of trials, N_t, is sufficiently large at each iteration, a potentially necessary connection with $w_i^c = 0$ for $x_i = 1$ can be selected correctly, and thus this necessary connection will be formed.

4 Concept Formation Networks

Here a self-organizing algorithm to get the solutions given by Eqs. (6) and (7) is proposed. Let us consider the unipolar binary case for features of the concept vector $\boldsymbol{x}$.

1. Initialize iteration $k = 0$, the number of neurons $n(0) = 1$, and the connection vector $||\boldsymbol{w}^1(0)|| = 0$.
2. Select one concept vector $\boldsymbol{x}^{i(k)} \in S$, $i(k) \in \{1, 2, \ldots, N\}$ and input it to the neural network.
3. Select a candidate neuron for the concept $\boldsymbol{x}^{i(k)}$ by the method described in Section 3.2.
4. Self-organizing the weights according to the the following condition

Case 1: If the candidate output y_c is equal to 1, try to create a new *awareness* weight w_i^c so that $\hat{x}_i(k) = 1$ by the extended Hebbian rule given in Eq. (19).

Case 2: If $0 < y_c < 1$, create new neuron $n(k) + 1$ with the same weight vector $\boldsymbol{w}^{n(k)+1} = \boldsymbol{w}^c$ and $n(k) + 1 \longrightarrow n(k)$. Then $c = n(k)$ and learn the weight to delete unnecessary connections by using Eq. (17).

Case 3: In the case of $y_c = 0$ if there is a neuron without any connections, then select this neuron for a candidate. Otherwise, create a new candidate neuron and $n(k) + 1 \longrightarrow n(k)$. Then try to create a randomly selected new *awareness* weight for this candidate by Eq. (19).

5. $n(k+1) = n(k)$, $k \longrightarrow k + 1$ and return to Step 2.

A concept formation ability of the self-organizing network is summarized in the following theorem.

Theorem 1 (Concept Formation Neural Structure Theorem) For a unipolar binary concept set $S = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^N\}$, there exists a self-organizing network with M neurons, $(M \geq N)$, such that

$$\boldsymbol{x}^i = \boldsymbol{w}^{j_i}, \quad i = 1, 2, \ldots, N, \quad j_i \in \{1, 2, \ldots, M\} \quad (20)$$

Proof of this theorem is straightforward and will be given in the oral presentation.

In addition to this ability, a major difference between the SOM and the proposed method given in Eqs. (14) and (16) is completeness of the input information. SOM supposes that the complete information of the input $\boldsymbol{x}$ can be given, while only an incomplete observation of the input, $\hat{\boldsymbol{x}}$, is possible to use in the proposed method. In the sense of a hypothesis that human cognition is based on an incomplete information, the proposed network can provide better model of human cognition.

5 Conclusion

We have developed a new self-organizing network model of concept formation by using neurons with dynamic and spatial changing weights. The proposed network can recognize concepts using incomplete information of concepts at current degree of understanding and develop its synaptic connections of inner informational structure to discover a new information of concepts and finally the complete information. The extended Hebbian rule proposed for structural adaptation is a possible informational formulation of a biologically evidential perspective in which some signal-driven special chemical matters can lead to growing directions of synaptic connections [6].

Acknowledgment

A part of this work was supported by The Ministry of Education, Culture, Sports, Science and Technology under Grant-in-Aid for Scientific Research No.15700119, and by Japan Society for the Promotion of Science under Grant-in-Aid for Scientific Research No.15300152.

References

[1] M. M. Gupta, L. Jin, and N. Homma, *Static and Dynamic Neural Networks – From Fundamentals to Advanced Theory* –, IEEE Press & Wiley, Hoboken, NJ, 2003.

[2] D. O. Hebb, *The Organization of Behavior*, Wiley, New York, 1949.

[3] T. Kohonen, *Self-Organization and Associative Memory*, 3rd ed., Springer-Verlag, 1989.

[4] N. Homma and M. M. Gupta, "Incremental Neural Learning by Dynamic and Spatial Changing Weights," *Proc. 15th IFAC World Congress*, vol. 3e, T-We-A04-2, 2002.

[5] N. Homma and M. M. Gupta: "A Self-Organizing Neural Structure for Concept Formation from Incomplete Observation," *Proc. IJCNN 2003*, pp. 2615–2618, 2003.

[6] K. Kohara, A. Kitamura, M. Morishima and T. Tsumoto, "Activity-Dependent Transfer of Brain-Derived Neurotrophic Factor to Postsynaptic Neurons," Science, vol. 291, pp. 2419–2423, 2001.

SANWA INSATSU SYUPPAN Co. Ltd.
4323-22 1Takaenishi, Oita, 870-1117, Japan
Tel : 097-596-7700 Fax : 097-596-7888
E-mail : sanwa@d-sanwa.com
http://www.d-sanwa.com