RAB

PROCEEDINGS OF THE EIGHTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 8th '03)

Vol. 1

Jan. 24-Jan. 26, 2003
B-Com Plaza, Beppu, Oita, JAPAN

Proceedings of The Eighth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 8th '03)

for Cognitive and Behavioral Intelligent Artificial Liferobot-2

January 24-26, 2003
B-Con Plaza, Beppu, Oita, Japan

Editors: Masanori Sugisaka and Hiroshi Tanaka

THE EIGHTH INTERNATIONAL SYMPOSIUM
ON
ARTIFICIAL LIFE AND ROBOTICS

(AROB 8th '03)

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)

COOPERATED BY

ISICE, IEICE, IEEE Japan Council, JARA

SUPPORTED BY

Ministry of Education, Culture, Sports, Science and Technology,
Japanese Government
Kyushu Bureau of Economy, Trade and Industry
Oita Prefecture
OITA CITY
Beppu City
Kyodo News
Jiji Press Ltd.
OITA GODO SHINBUNSYA
The Asahi shimbun
The Mainichi Newspapers
THE YOMIURI SHIMBUN
The Nishinippon Newspaper Co.
THE NIKKAN KOGYO SHINBUN. LTD
Nihon Keizai Shimbun, INC.
NHK (Japan Broadcasting Corporation)
OITA BROADCASTING SYSTEM, INC.
Television Oita System
Oita Asahi Broadcasting Co. Ltd.
Oita Prefectural Industrial Organization
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)

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)
K.Kosuge (Tohoku University, Japan)
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 Okulahoma , 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 (RIKEN, 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 (Kyusyu 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 (Miyazaki University, 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)
H.Miyagi (Ryukyu University, Japan)
M.Nakamura (Saga University, Japan)
R.Nakatsu (KWANSEI GAKUIN University, Japan)
S.Omatsu (University of Osaka Prefecture, Japan)
M.Okamoto (Kyusyu 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 (MITI, MEL, Japan)
K.Shimohara (ATR, Japan)
M.Sugisaka (Oita University, Japan)
H.Tanaka (Tokyo Medical & Dental University, Japan)
   (Chairman)
M.Tanaka-Yamawaki (Miyazaki University, Japan)
N.Tosa (ATR, Japan)
K.Ueda (The University of Tokyo, Japan)
K.Uosaki (Tottori University, Japan)
H.Wakamatsu (Tokyo Medical & Dental University, Japan)

K.Watanabe (Saga University, Japan)
M.Yano (Tohoku University, Japan)

LOCAL ARRANGEMENT COMMITTEE

T.Kubik (Oita University, Japan)
K.B.Kubik (Oita University, Japan)
A.Loukianov (Oita University, Japan)
K.Shibata (Oita University, Japan)
M.Sugisaka (Oita University, Japan) (Chairman)
Y.Suzuki (Tokyo Medical & Dental University, Japan)
X.Wang (Oita Institute of Technology, Japan)
F.Xiongfeng (Oita University, Japan)

HISTORY

This symposium was founded in 1996 by the support of Science and International Affairs Bureau, Ministry of Education, Science, Sports and Culture, Japanese Government. Since then, this symposium was held every year at B-Con Plaza, Beppu, Oita, Japan except Tokyo, Japan (AROB 6th '02). The Eighth symposium will be held on 24-26 January 2003, 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 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
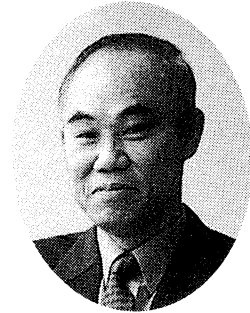Robotics
Robust virtual engineering
Virtual reality

COPYRIGHTS

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 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.

M. Sugisaka

December 20, 2002

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 Eighth International Symposium on Artificial Life and Robotics (AROB 8th '03). 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, 2002

TIME TABLE

	RoomA	RoomB	RoomC	RoomD
1/23(Thu.) 8:00				
13:00				
17:00	Registration (Registration Desk)			
	Welcome Party(in Hotel Arthur 10th Floor)			
1/24(Fri.) 8:00	Registration (Registration Desk)			
9:00	Chair A. Loukianov GS1(6)	Chair T. Kitazoe IS1(6)	Chair W. Wells IPS1(3)	
10:30				
10:40				Opening Ceremony
11:00				Plenary Talk Chair J. Casti PT1 M.Sugisaka
12:00	Lunch			
13:00				
14:00	Chair T. Arita GS2(5)	Chair J. Johnson GS3(5)	Chair H. Kang GS4(5)	
14:15				
	Chair X. Feng GS5(4)	Chair A. Ohuchi GS6(3)	Chair H. Hirayama GS19(4)	
15:15				
15:30				
	Chair J. Casti GS7(4)	Chair T. Hoya GS8(4)	Chair T. Kubik GS9(4)	
16:30				
	Chair K. Shibata GS10(4)	Chair H. Suzuki IS3(4)	Chair H.Tanaka GS11(3)	
17:30				

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

GS1 Robotics-I
GS2 Multi-agent systems
GS3 Bioinformatics, Molecular Biology & Brain Science
GS4 Intelligent Control and Modeling-I
GS5 Genetic Algorithms & Evolutionary Computations-I
GS6 Artificial Intelligence
GS7 Virtual Reality & Economic Data Mining
GS8 Computer and Robot Vision-I
GS9 Fuzzy Control
GS10 Reinforcement Learning-I
GS11 Image Processing & Pattern Recognition-I
GS12 Reinforcement Learning-II
GS13 Image Processing & Pattern Recognition-II
GS14 Micro-Robot World Cup Soccer Tournament &
                     Micromachines
GS15 Robotics-II
GS16 Mobile Vehicle-I
GS17 Neural Networks
GS18 Intelligent Control and Modeling-II
GS19 Genetic Algorithms & Evolutionary Computations-II
GS20 Computer and Robot Vision-II
GS21 Mobile Vehicle-II

GS22 Artificial Life
GS23 Computer & Internet Security
IS1 Robot Control and Image Processing
IS2 Neural Network Applications
IS3 Evolution of codes, Behaviors, and networks
IS4 Artificial Chemistry-I
IS5 Artificial Chemistry-II
IS6 Welfare and Medical Engineering
IS7 Nonlinear Modeling and its Applications
IS8 Machine Intelligence and Robotic Control
IS9 Neural Network
IS10 Genetic Algorithms for Engineering Optimization
IS11 Interaction and Intelligence
IS12 Artificial Brain
IS13 Artificial Mind
IS14 Genetic Algorithms for Production and Distribution
IS15 Artificial Life and Application
IS16 Soft Robotics and Information
IS17 Applications for Intelligent Control
IPS1 Invited Professor's Session (I)
IPS2 Invited Professor's Session (II)
IPS3 Invited Professor's Session (III)

	RoomA	RoomB	RoomC	RoomD
1/25(Sat.) 8:00	Registration (Registration Desk)			
9:00	Chair H. Suzuki IS4(4)	Chair J.M. Lee GS12(4)	Chair K. Ida IS10(4)	
10:00	Chair H. Suzuki IS5(3)	Chair M. Nakamura GS14(3)	Chair T. Ishimatsu IS6(3)	
10:45	Coffee Break			
11:00				Plenary Talk Chair Y. Zhang PT2 T. Fukuda
12:00				
13:00	Lunch			
	Chair S. Omatsu GS15(6)	Chair C. Zhang GS17(5)	Chair Z. Bubnicki IPS2(2)	
14:30 / 14:45	Coffee Break			
	Chair M. Gen IS14(5)	Chair H. Sayama GS16(6)	Chair I. Yoshihara IS9(5)	
16:00				
16:15	Chair X. Wang GS13(3)	Chair H. Hashimoto IS11(5)	Chair A. Buller IS12(5)	
17:00				
17:15				
18:00	AFOB Award Ceremony (Chair K. Watanabe)			
18:20	Banquet (Hotel Shiroyama)			
20:30				

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

GS1 Robotics-I
GS2 Multi-agent systems
GS3 Bioinformatics, Molecular Biology & Brain Science
GS4 Intelligent Control and Modeling-I
GS5 Genetic Algorithms & Evolutionary Computations-I
GS6 Artificial Intelligence
GS7 Virtual Reality & Economic Data Mining
GS8 Computer and Robot Vision-I
GS9 Fuzzy Control
GS10 Reinforcement Learning-I
GS11 Image Processing & Pattern Recognition-I
GS12 Reinforcement Learning-II
GS13 Image Processing & Pattern Recognition-II
GS14 Micro-Robot World Cup Soccer Tournament &
      Micromachines
GS15 Robotics-II
GS16 Mobile Vehicle-I
GS17 Neural Networks
GS18 Intelligent Control and Modeling-II
GS19 Genetic Algorithms & Evolutionary Computations-II
GS20 Computer and Robot Vision-II
GS21 Mobile Vehicle-II

GS22 Artificial Life
GS23 Computer & Internet Security
IS1 Robot Control and Image Processing
IS2 Neural Network Applications
IS3 Evolution of codes, Behaviors, and networks
IS4 Artificial Chemistry-I
IS5 Artificial Chemistry-II
IS6 Welfare and Medical Engineering
IS7 Nonlinear Modeling and its Applications
IS8 Machine Intelligence and Robotic Control
IS9 Neural Network
IS10 Genetic Algorithms for Engineering Optimization
IS11 Interaction and Intelligence
IS12 Artificial Brain
IS13 Artificial Mind
IS14 Genetic Algorithms for Production and Distribution
IS15 Artificial Life and Application
IS16 Soft Robotics and Information
IS17 Applications for Intelligent Control
IPS1 Invited Professor's Session (I)
IPS2 Invited Professor's Session (II)
IPS3 Invited Professor's Session (III)

	RoomA	RoomB	RoomC	RoomD
1/26(Sun.) 8:00	Registration (Registration Desk)			
9:00	Chair K. Shimohara IS13(5)	Chair K. Aihara IS7(5)	Chair Y. Zhang IPS3(2)	
10:30 / 10:45				
10:45 / 12:00	Chair B. Price GS18(4)	Chair S. Omatsu IS2(4)	Chair J. Wang GS20(5)	
13:00	Chair H. H. Lund GS21(5)	Chair K. Watanabe IS8(5)	Chair T.X. Yan GS22(3)	
13:45			Chair P. Sapaty GS23(3)	
14:15		Chair T. Yamamoto IS16(4)		
14:30	Chair K-B. Sim IS15(4)		Chair J.J. Lee IS17(4)	
15:30	Farewell Party (Room D)			
16:30				

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

GS1 Robotics-I
GS2 Multi-agent systems
GS3 Bioinformatics, Molecular Biology & Brain Science
GS4 Intelligent Control and Modeling-I
GS5 Genetic Algorithms & Evolutionary Computations-I
GS6 Artificial Intelligence
GS7 Virtual Reality & Economic Data Mining
GS8 Computer and Robot Vision-I
GS9 Fuzzy Control
GS10 Reinforcement Learning-I
GS11 Image Processing & Pattern Recognition-I
GS12 Reinforcement Learning-II
GS13 Image Processing & Pattern Recognition-II
GS14 Micro-Robot World Cup Soccer Tournament & Micromachines
GS15 Robotics-II
GS16 Mobile Vehicle-I
GS17 Neural Networks
GS18 Intelligent Control and Modeling-II
GS19 Genetic Algorithms & Evolutionary Computations-II
GS20 Computer and Robot Vision-II
GS21 Mobile Vehicle-II

GS22 Artificial Life
GS23 Computer & Internet Security
IS1 Robot Control and Image Processing
IS2 Neural Network Applications
IS3 Evolution of codes, Behaviors, and networks
IS4 Artificial Chemistry-I
IS5 Artificial Chemistry-II
IS6 Welfare and Medical Engineering
IS7 Nonlinear Modeling and its Applications
IS8 Machine Intelligence and Robotic Control
IS9 Neural Network
IS10 Genetic Algorithms for Engineering Optimization
IS11 Interaction and Intelligence
IS12 Artificial Brain
IS13 Artificial Mind
IS14 Genetic Algorithms for Production and Distribution
IS15 Artificial Life and Application
IS16 Soft Robotics and Information
IS17 Applications for Intelligent Control
IPS1 Invited Professor's Session (I)
IPS2 Invited Professor's Session (II)
IPS3 Invited Professor's Session (III)

TECHNICAL PAPER INDEX

January 24 (Friday)

Room D

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

January 25 (Saturday)

11:00~12:00 PT-2 Plenary Talk2
Chair Y. Zhang

January 24 (Friday)

Room A

9:00~10:30 GS1 Robotics-1
Chair: A. Loukianov (Oita University, Japan)

13:00~14:15 GS2 Multi-agent systems
Chair: T. Arita (Nagoya University, Japan)

14:15~15:15 GS5 Genetic Algorithms & Evolutionary Computation- I
Chair: X. Feng (Oita University, Japan)

15:30~16:30 GS7 Virtual Reality & Economic Data Mining
Chair: J. Casti (SFI, USA)

16:30~17:10 GS10 Reinforcement Learning- I
Chair: K. Shibata (Oita University, Japan)

for unobservable dynamic environment
  K. Umesako, M. Obayashi, K. Kobayashi (Yamaguchi University, Japan)

Room B

9:00~10:30 IS1 Robot Control and Image Processing
Chair: T. Kitazoe (Miyazaki University, Japan)
Co-Chair: M. Tabuse (Miyazaki University, Japan)

13:00~14:15 GS3 Bioinformatics, Molecular Biology & Brain Science
Chair: J. Johnson (Open University, UK)

14:15~15:00 GS6 Artificial Intelligence
Chair: A. Ohuchi (Hokkaido University, Japan)

15:30~16:30 GS8 Computer and Robot Vision- I
Chair: T. Hoya (BSI RIKEN, Japan)

16:30~17:30 IS3 Evolution of codes, Behaviors, and network
Chair: H. Suzuki (ATR Human Information Science Labs., Japan)
Co-Chair: K. Shimohara (Kyoto University, ATR, Japan)

Room C

9:00~10:30 ISP1 Invited Professor's Session(I)
Chair: W. Wells (University of Nevada-Las Vegas, USA)

13:00~14:15 GS4 Intelligent Control and Modeling- I
Chair: H. Kang (Chung- Ang University, Korea)

14:15~15:15 GS19 Genetic Algorithms & Evolutionary Computations- II
Chair: H. Hirayama (Asahikawa Medical College, Japan)

15:30~16:30 GS9 Fuzzy Control
Chair: T. Kubik (Oita University, Japan)

16:30~17:30 GS11 Image Processing & Pattern Recognition- I
Chair: H. Tanaka (Tokyo Medical and Dental University, Japan)

January 25 (Saturday)

Room A

9:00~10:00 IS4 Artificial Chemistry- I

Chair: H. Suzuki (ATR Human Information Science Labs, Japan)
Co-Chair: J-Q. Liu (ATR Human Information Science Labs, Japan)

10:00~10:45 IS5 Artificial Chemistry-II
Chair: H. Suzuki (ATR Human Information Science Labs, Japan)
Co-Chair: J-Q. Liu (ATR Human Information Science Labs, Japan)

13:00~14:30 GS15 Robotics-II
Chair: S. Omatsu (Osaka Prefecture University, Japan)

K. Wada, T. Shibata, T. Saito, K. Tanie (AIST, Japan)

14:45~16:00 IS14 Genetic Algorithms for Production and Distribution
Chair: M. Gen (Ashikaga Institute of Technology, Japan)
Co-Chair: M. Sasaki (Ashikaga Institute of Technology, Japan)

16:00~17:15 GS13 Image Processing & Pattern Recognition-Ⅱ
Chair: X. Wang (Oita Institute of Technology, Japan)

Room B

9:00~10:00 GS12 Reinforcement Learning-Ⅱ
Chair: J.M. Lee (Pusan National University, Korea)

10:00~10:45 GS14 Micro-Robot World Cup Soccer Tournament & Micromachines
Chair: M. Nakamura (Saga University, Japan)

13:00~14:30 GS17 Neural Networks
Chair: C. Zhang (Tsinghua University, P.R.China)

T. Hoya (BSI RIKEN, Japan)

14:45~16:00 GS16 Mobile Vehicle- I
Chair: H. Sayama (University of Electro-Communication, Japan)

16:00~17:15 IS11 Interaction and Intelligence

Chair: H. Hashimoto (The University of Tokyo, Japan)

Room C

9:00~10:00 IS10 Genetic Algorithms for Engineering Optimization
Chair: K. Ida (Maebashi Institute of Technology, Japan)
Co-Chair: T. Yokota (Ashikaga Institute of Technology, Japan)

10:00~10:45 IS6 Welfare and Medical Engineering
Chair: T. Ishimatsu (Nagasaki University, Japan)

Japan)

13:00~14:30 IPS2 Invited Professor's Session(II)
Chair: Z. Bubnicki (Wroclaw University of Technology, Poland)

14:45~16:00 IS9 Neural Network
Chair: I. Yoshihara (Miyazaki University, Japan)
Co-Chair: M. Yasunaga (University of Tsukuba)

16:00~17:15 IS12 Artificial Brain

Chair: A. Buller (ATR Human Information Science Labs, Japan)
Co-Chair: K. Shimohara (ATR Human Information Science Labs, Japan)

January 26 (Sunday)

Room A

9:00~10:30 IS13 Artificial Mind
Chair: K. Shimohara (ATR Human Information Science Labs, Japan)
Co-Chair: H. Kozima (Communications Research Laboratory, Japan)

J. Liu (Central South University, P.R.China)

10:45~12:00 GS18 Intelligent Control and Modeling- II
Chair: B. Price (Open University, UK)

13:00~14:30 GS21 Mobile Vehicle- II
Chair: H.H. Lund (University of Southern Denmark, Denmark)

M. Sugisaka (Oita University, RIKEN, Japan)

14:30~15:30 IS15 Artificial Life and Application
Chair: K.B. Sim (Chung-Ang University, Korea)

IS15-1 *Swarm behavior of multi-agent system based on artificial immune system* ······561
     K.B. Sim, D.W. Lee (Chung-Ang University, Korea)

IS15-2 *Bayesian clustering for determination of dynamic web preference* ······565
     D.S. Kim (HanShin University, Korea)
     J.H. Choi (Kimpo College, Korea)
     M.S. Han (ETRI, Korea)

IS15-3 *The research about growth and behavior of a virtual life by* ······569
   using genetic algorithm
     M.S. Kwon, D.W. Kim, J.Y. Lee, H. Kang (Chung-Ang University, Korea)

IS15-4 *Structure identification of neuro-fuzzy models using genetic algorithms* ······573
     B.H. Wang (Kangnung National University, Korea)
     H.J. Cho (Purdue University, USA)

Room B

9:00~10:30 IS7 Nonlinear Modeling and its Applications
Chair: K. Aihara (The University of Tokyo, Japan)
Co-Chair: H. Suzuki (The University of Tokyo, Japan)

IS7-1 *Complex behavior of a simple partial discharge model* ······577
     H. Suzuki, K. Aihara (The University of Tokyo, Japan)
     S. Ito (Kanazawa University, Japan)

IS7-2 *Dimensional analysis of the Hodgikin-Huxley equations with noise:* ······581
   Effects of noise on chaotic neurodynamics
     H. Tanaka, K. Aihara (The University of Tokyo, Japan)

IS7-3 *Encoding ternary information using a chaotic neural network* ······585
     J.K. Ryeu (Dongyang University, Korea)

IS7-4 *Human-like decision making in an autoassociative neural network* ······589
   with dynamic synapses
     Z. Wang (DongHua University, P.R.China)
     K. Aihara (The University of Tokyo, Japan)

IS7-5 *Origins of stochasticity in gene expression and control of the fluctuation* ······593
     Y. Morishita, K. Aihara (The University of Tokyo, Japan)

10:45~12:00 IS2 Neural Network Applications

Chair: S. Omatsu (Osaka Prefecture University, Japan)
Co-Chair: A. Selamat (Osaka Prefecture University, Japan)

13:00~14:15 IS8 Machine Intelligence and Robotic Control
Chair: K. Watanabe (Saga University, Japan)
Co-Chair: K. Izumi (Saga University, Japan)

14:15~15:30 IS16 Soft Robotics and Information
Chair: T. Yamamoto (University of the Ryukyus, Japan)
Co-Chair: H. Kinjo (University of the Ryukyus, Japan)

Room C

9:00~10:30 IPS3 Invited Professor's Session(Ⅲ)
Chair: Y. Zhang

10:45~12:00 GS20 Computer and Robot Vision-Ⅱ
Chair: J. Wang (Oita University, Japan)

13:00~13:45 GS22 Artificial Life
Chair: T.X. Yan (University of Science & Technology Beijing, P.R.China)

13:45~14:30 GS23 Computer & Internet Security
Chair: P. Sapaty (Oita University, Japan)

14:30~15:30 IS17 Applications for Intelligent Control
Chair: J.J. Lee (Korea Advanced Institute of Science and Technology, Korea)

A new artificial life body
-biologically inspired dynamic bipedal humanoid robots -

Masanori Sugisaka[1,2], Kouta Imamura[1], Kouji Tokuda[1] and Maoki Masuda[1]
[1]Department of Electrical and Electronic Engineering
The University of Oita
700 Oaza Dannoharu, Oita, 870-1192, Japan
tel 001-81-97-554-7831 fax 001-81-97-554-7841
msugi@cc.oita-u.ac.jp http://arob.cc.oita-u.ac.jp
[2]Bio-Mimetic Control Research Center
The Institute of Physical and Chemical Research(RIKEN) at Nagoya
2271-130, Anagahora, Shimoshidami, Moriyama-ku, Nagoya, 463-0003, Japan

Abstract-Recently biologically inspired bipedal dynamic humanoid robot was developed at Artificial Life and Robotics laboratory of Oita University. This bipedal humanoid robot is able to walk dynamically and to go up and down stairs. The central pattern generator developed produces various kinds of walking patterns. This robot has a pair of small CMOS color CCD cameras, speaker and mike in the head part and will have GPS, portable phone, and other sensors in the body part so that the integration of locomotion and behavior to achieve specific demonstrations is realized in this bipedal humanoid robot. This project develops dynamic mobility and the ability for autonomous recognition and navigation using the biological central nervous system, the brain system, and the real time control system. Also the design principles that demonstrate the dynamic interaction between neural and mechanical controls will be clarified. In Phase I the platform of a small size bipedal humanoid robot is used to develop autonomous locomotion and autonomous sensing and navigation. In Phase II of the project the iteration on platform design for human size bipedal humanoid robots will be performed for operational testing. The development of bipedal humanoid robots that captures biological systems with unique principles and practices could dramatically increase their performance in tasks for national security needs.

1. Introduction

The biologically inspired bipedal humanoid robot developed recently at Artificial Life and Robotics Laboratory of Oita University is considered as a new artificial life body with the artificial brain[1]-[5] which has to provide the robot with abilities to perceive an environment, to interact with humans, to make intelligent decisions and to learn new skills. On the other hand, we have already developed an artificial brain for an artificial life robot "Tarou"[6]. These two types of artificial brains have different specifications. The artificial brain developed in [1]-[5] consisted of the hardware neural network (Ricoh neurocomputer RN-2000), interfaces, and 32 bits microcomputer. The artificial brain installed in Tarou consists of software neural networks.

In order for the bipedal humanoid robot to behave autonomously for achieving its tasks, the robot requires a brain system like living creatures. The brain system is a complex system which has functions such as control, recognition, learning, and other higher functions. The relation between the brain and outside world is given in Fig.1.

The brain has five functions and 1,000 billions neurons;

(1) recognition of outside world (five senses),
(2) exercise control (conscious control; walking and balancing, unconscious control; playing violin and piano),
(3) consciousness (level of consciousness in awake is higher than level of consciousness in sleep),
(4) emotion or passion(open the pupil, whet the nail, erect hair)
(5) memorization and learning.
The five functions are located locally in the brain system and, hence, we feel that there exists a heart in living things as a whole, namely, there exists life.

The central nervous system which consists of both the brain system and the spinal cord is shown in Fig.2.

In this paper, at first, the concept of artificial brain is introduced briefly. Secondly, the brief description of the biologically inspired dynamic bipedal humanoid robot is given with the specification of the robot. Thirdly, experimental results are illustrated. Finally in conclusions we show the problems to be solved for the special purpose which

achieves a given task.

output

⇑ outside world

centrifugal
neuron

muscles·exercise organs effectors (interfaces)

⇑

centripetal
neuron

brain (neural networks)

⇑

sensory organs (interfaces)				
sense of vision	auditory sense	sense of taste	sense of touch	sense of smell

⇑ life

input
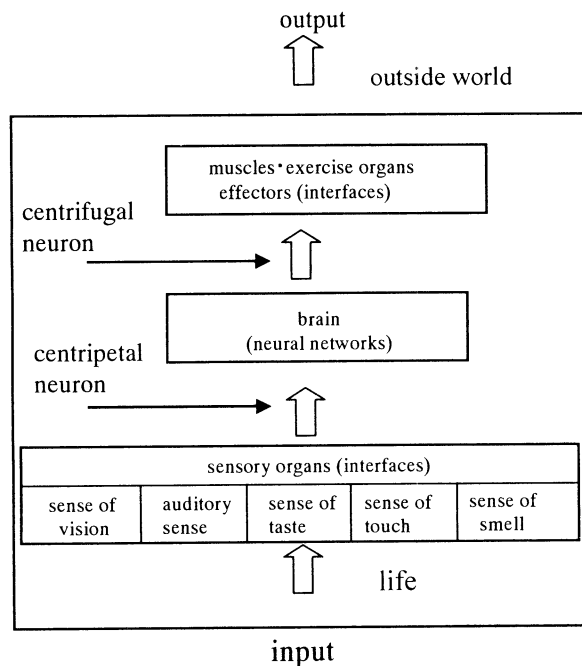
Fig.1 Relation between brain and outside world

cerebrum

diencephalon

cerebellum

encephalon

brain
stem

mesencephalon

metencephalon
spinal code

pons

five senses

1,000 billion neurons

* central nervous system
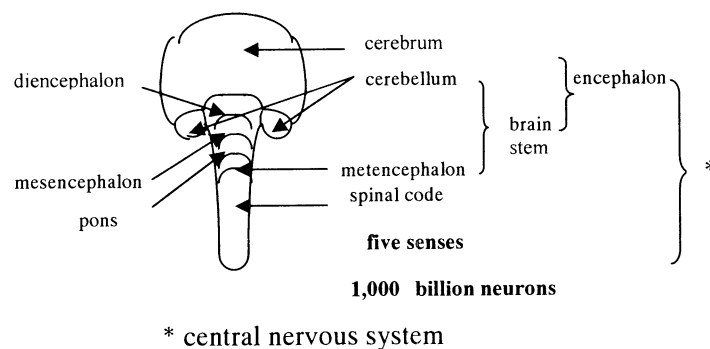
Fig.2 Function of brain and central nervous system

2. Artificial brain

The artificial brain for Tarou consists of the special softwares. The hardware is shown in Fig.3 where two CCD cameras, two computers, two DC moters, one stepping motor, six ultra sonic sensors, 5 LEDs are installed. One portable computer is used for voice recognition and command executing system (however, this computer is replaced by the above two computers later). The artificial brain is able to perform vision based navigation, voice recognition, and face detection and face recognition. In Fig.3 the hardware structure is shown and the software structure is given in Fig. 4. In Fig. 5 behavioral table is illustrated.
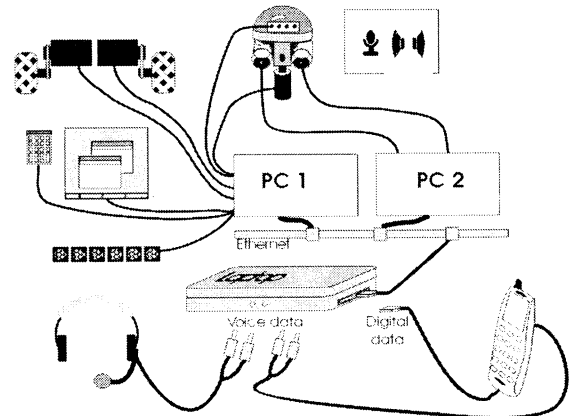
Fig. 3 Hardware structure of Tarou
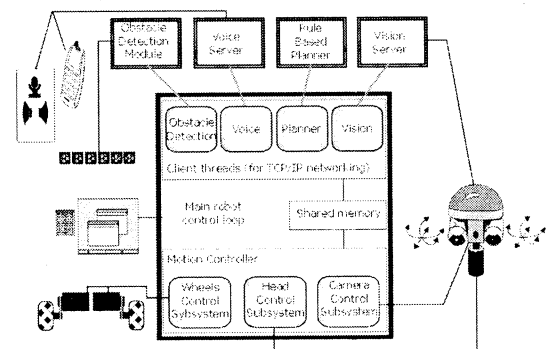
Fig. 4 Software structure of artificial brain for Tarou
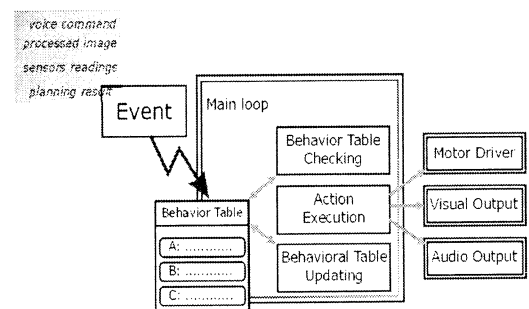
Fig. 5 Behavioral table

2.1 Vision based navigation

There are two types of landmarks used in the navigation based on two CCD video cameras which can capture nearly 24 images/s: one is the continuous landmark (Guideline), another is the common landmarks (Circle, triangle, etc). The guideline is used for positioning, and other landmarks can make the robot perform some specific action, such as turn, stop etc. The robot moves at the least speed of 12cm/s. After one image was taken, the guideline and landmarks were extracted and recognized from that image. The robot will act based on the results of the image processing. And it has the capability to search its target as fast as possible when it is moving.

2.2 Voice recognition

Our robot has a voice recognition and command executing system, which are distributed on two computers on hardware and link and local network. The voice recognition system utilizes IBM ViaVoice SDK (Software Developer's Kit) for dictation and IBM ViaVoice TTS (Text to Speech) SDK for speech application.

By using this system, our robot can now recognize the following commands in English:
1 Go ### meter(s)/centimeter(s)
2 Turn your head to the left/right
3 Straighten your head
3 Look up/down/straight
4 Turn left/right ### degrees
5 No (for canceling of a command)
(### means a number)
IBM ViaVoice provides a tool for building a personal voice model. For example, if a voice model is made by a Japanese people (who speaks English with Japanese accent), then, it will be easier for the voice recognition system to understand other Japanese people's speaking.

2.3 Face detection & face recognition

We have built a real-time face recognition system which is combined with face detection (based on a neural network) and face recognition (based on Embedded Hidden Markov Models (EMM)) techniques. It constantly takes images from the surroundings with the camera mounting on the robot's head, finds faces in them. If a face is detected, the system tries to recognize it.

The neural network is trained with many face examples to get the concept of a face. It receives a 20×20 pixel region of the image as input, and generates an output ranging from 0 to 1, signing how close it is to a face. We use it to examine a sub-image subtracted

from the input image whether it is a face.
For face recognition, we built a separate EMM model for each person in the face database. If a face is detected, it will be matched against each model in the database. The model, which yields the largest similarity value, is reported as the host.
The system have achieved a detection rate of more than 85% for front views or slightly rotated views of faces.

These techniques of artificial brain used in Tarou are easily applicable for designing an artificial brain for a biologically inspired dynamic bipedal humanoid robot (BiOMAN-1).

3. Biologically inspired dynamic bipedal humanoid robot (BiOMAN-1)

3.1 Specification

The specification of the BiOMAN-1 developed at our laboratory is illustrated in Table 1. The parts which compose BiOMAN-1 is sold in stores and is easily obtained so that the cost is relatively cheep(less than ¥200,000 including several sensors shown in Table 1). The shape was designed by ourselves. The arrangement of target computer and other parts were all made by our hands.

The freedom of body is shown in Fig. 6. At this stage, arms and hands are not mounted due to the delay of arrival of small servomotors with brackets. We will mount these small servomotors when they arrived.
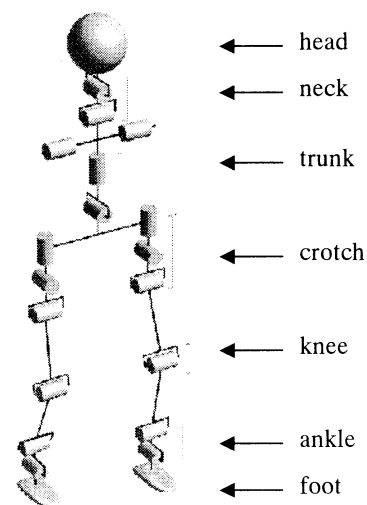
Fig.6 Freedom of body

3.1 Experimental results

In order for the BiOMAN-1 to walk statically and

dynamically, the home position have to be decided beforehand. One example used in our experiments is shown in Fig. 7 where M** denotes servomotor's number and the numerical values show the one home position. Based on the home position, various walking patterns were developed according to the biological fixed walking (moving) pattern generated from the central pattern generator (CPG) genetically equipped in the central nervous system in the brain. The results will be shown by video.

M00	
0	
M01	
0	
M10	
135	
M11	
125	

M12		M13
85		96
M14		M19
97		102
M15		M20
98		116
M16		M21
142		28
M17		M22
120		36
M18		M23
96		42

left leg   right leg

M**: motor number

4. Conclusions

In this paper, brief introductions for both the artificial brains for the artificial life robot Tarou and the biologically inspired dynamic bipedal humanoid robot (BiOMAN-1) in Phase I research are given. The artificial brain developed in Tarou will be applied to BiOMAN-1 soon that enables to behave in all terrain autonomously. These researches are our next objective to be developed.

References

[1]M. Sugisaka, "Intelligent control and learning using artificial brain constructed by neurocomputer," Intelligent Automation and Control, Recent Trends in Development and Applications, Vol.4, pp.685-689, TSI Press Series, 1996, 3.

[2]M.Sugisaka, "Neurocomputer control in an artificial brain for tracking moving objects," Artificial Life and Robotics, Vol.1, No.1, pp.47-51, 1997, 8.

[3]M. Sugisaka, N. Tonoya and T. Furuta, "Neural networks for control in an artificial brain of recognition and tracking system," Artificial Life and Robotics, Vol.2, No.3, pp.119-122, 1998.

[4]M. Sugisaka, "Design of an artificial brain for robots," Artificial Life and Robotics, Vol.3, No.1, pp.7-14,1999.

[5]M. Sugisaka, X. Wang and J.J. Lee, "Artificial brain for a mobile vehicle," Applied Mathematics and Computation, Vol.111, No.2&3, pp.137-145, 2000,5.

[6]C. A. Radix, A. A. Loukianov, M. Sugisaka, "Evaluating Motion on the Alife Robot Prototype", Proc. of the 32nd International Symposium on Robotics(ISR 2001), pp.714-719, Seoul, Korea, April 19-21, 2001

Specification		
size	height 520 mm	
weight	5.5kg	
actuators	·KO puropo, servomotor PDS-2144FET (torque, 13kg-cm): 12 ·Futaba, servomotor S3003 (torque, 3.2kg-cm): 4	
sensors	foot part	not decided
	body part	·Omron, linear inclination sensor D5R-L02-60: 2 ·Sharp, distance sensor GP2D12-PSD:2 ·S.T.L.Japan, electric gyroscope HS-EG3(to be mounted): 1
	each joints	not decided
	head part	·C-MOS CCD camera CAM25C: 2 ·UHF wireless video transmitter TR-25R: 2 ·UHF wireless video receiver CX-135R: 2 ·data converter DAC-100: 2 to be mounted
tool for developing software for target computer	host computer	personal computer (Window XP, etc.)
controller	target computer	HSWB01

Table 1 Specification of the biologically inspired dynamic humanoid robot (BiOMAN-1)

Intelligent Robot as an Artificial Living Creature

Toshio Fukuda and Yasuhisa Hasegawa

Department of Micro System Engineering, Nagoya University
http://www.mein.nagoya-u.ac.jp

Abstract

This paper discusses the machine (system) intelligence from the viewpoints of learning, adaptation and evolution of living things. Next, this paper introduces examples of integrated system developed by industrial companies, those are designed to work domestic area with intelligence. Finally, this paper shows an example of intelligent robotic system: brachiation robot developed in our project.

1 Introduction

Intelligence for robot to grow and evolve can be observed both through growth in computational power, and through the accumulation of knowledge of how to sense, decide and act in a complex and dynamically changing environment. There are four elements of intelligence: sensory processing, world modeling, behavior generation and value judgement. Input to, and output from, intelligent system are via its body with sensors and actuators. Recently, intelligent systems have been discussed not only in knowledge engineering, computer science, mechatronics and robotics, but also psychology and brain science . Various methodologies about intelligence have been successfully developed.

Artificial intelligence (AI) builds an intelligent agent, which perceives its environment by sensors, makes a decision and takes an action [1]. McCulloch and Pitts suggested that suitably defined networks could learn [2], and furthermore, Newell and Simon developed general problem solver. Afterward, knowledge-based system including expert systems has been developed. In addition, language processing, reasoning, planning, and others have been discussed in AI. Human language enables the symbolic processing of information, and is translate into numerical information according to an objective, that is, word is classified into a certain attribute out of much information. In this way, the symbolic information processing have resulted in success in AI. Further, The recent research fields concerning intelligence, include brain science, soft computing, artificial life and computational intelligence [1]-[6].

Computational intelligence from the viewpoints of biology, evolution and self-organization tries to construct intelligence by internal description, while classical AI tries to construct intelligence by external (explicit) description. Therefore, information and knowledge of a system in computational intelligence should be learned

or acquired by itself.

Robot is required to have intelligence and autonomous ability when it works far from an operator with large time delay, or when it works in a world containing ambiguous information. The robot collects or receives necessary information concerning its external environment, and takes actions to the environment. Both of them are often designed by human operators, but ideally, the robot should automatically perform the given task without human assistance. Computational intelligence methods including neural network (NN), fuzzy logic (FL) and evolutionary computation (EC), reinforcement learning, expert system and others, have been applied to realize intelligence on the robotic systems [2]-[11]. In addition, behavior-based AI has been discussed as learning methods dependent on environmental information [1],[12]. The behavior-based AI stresses the importance of the interaction between robot and environment shown in Fig. 1, while classical AI is based on the representation and manipulation of explicit knowledge. Recently, behavior analysis and training as methodology for behavior engineering and model-based learning, have been proposed [12]-[14]. In this paper, we introduce a basic technique to build an intelligent system. After that, we introduce adaptation algorithm for brachiation robot.
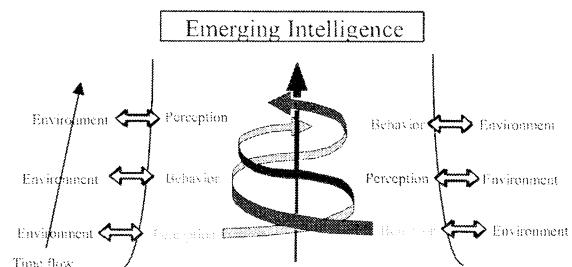
Figure 1: Emergence of intelligence

2 Intelligent System

Human being makes decision and takes actions based on the sensing information and internal state, when we consider human beings as an intelligent system. Furthermore, human beings can learn by acquiring or perceiving information concerning reward and penalty from the external environment. Thus, human beings perform the perception, decision making and action (Fig.2). In

future, a robot will work out of a factory which environment was simplified so that a robot could recognize it. The robot is required to have much intelligence and autonomous capability when it works far from an operator with large time delay such as tele-operation, when sensing information is contained ambiguous information. Key technologies for system intelligence and autonomous are knowledge representation, recognition, inference, search, planning, learning, prediction and so on[1].

The system intelligence emerges from the synthesis of various intelligent capabilities of the systems. Consequently, the whole intelligence of a system depends on the structure for processing information on hardware and software, and this means that the structure determines the potentiality of intelligence. Therefore, we should consider a whole structure of intelligence for information process flow over the hardware and software.
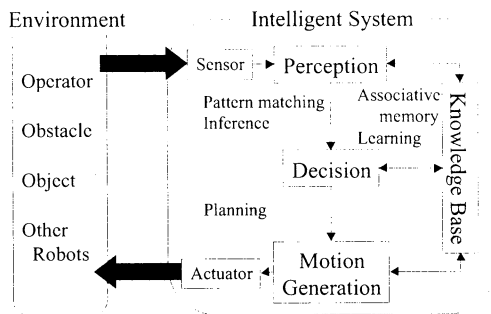
Figure 2: Interaction between intelligent system and its environment

To realize higher intelligent system, a synthesized algorithm of various techniques is required. Figure 3 shows the synthesis of NN, FL and EC. Each technique plays the peculiar role for intelligent function. There are not complete techniques for realizing all features of intelligence. Therefore, we should integrate and combine some techniques to compensate the disadvantages of each technique. The main characteristics of NN are to classify or recognize patterns, and to adapt itself to dynamic environments by learning, but the mapping structure of NN is a black box and incomprehensible. On the other hand, FL has been applied for representing human linguistic rules and classifying numerical information into symbolic class. It also has reasonable structure for inference, which is composed of if-then rules like human knowledge. However FL does not fundamentally have the learning capability. Fuzzy-neural networks have developed for overcoming their disadvantages [5]. In general, the neural network part is used for its learning, while the fuzzy logic part is used for representing knowledge. Learning capability is fundamentally performed as necessary change such as incremental learning, back propagation method and delta rule based on error functions. EC can also tune NN and FL. However, evolution can be defined as resultant or accidental change, not necessary change, since the EC can not predict and estimate the effect of the change. To summarize, an intelligent system can quickly adapt to dynamic environment by NN and FL with the back propagation method or delta rule, and furthermore, the structure of intelligent system can globally evolve by EC according to the objective problems. The capabilities concerning learning adaptation and evolution can construct more intelligent system. Intelligence arises from the information processing on the linkage of perception, decision making and action.
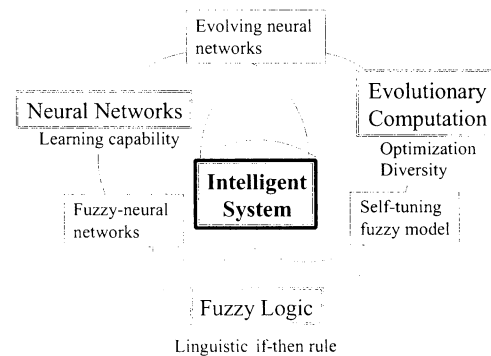
Figure 3: Synthesis of NN, FL and EC

3 Integrated system with intelligence

3.1 Intelligent robots for domestic work

Many industrial companies in Japan have announced the concept and prototype of new type of robots since several years. Figure 4–7 are pioneer works. It is thought that an intelligent robot will play an important role as to solve the social problems such as traffic, medical, welfare, domestic work and environment protection instead of human labor. In order to expand robots' working area out of an industrial factory, a robot should at least recognize its situation and environment including human correctly. The current computational intelligence does not have such a high recognition ability, therefore "AIBO" shown in fig. 6 is released as a pet robot in 1999 since pet robot is not generally required to precise recognition enough to perform practical work but it plays football game in RoboCup. On the other hand, humanoid robot with same dimensions as human being has advantage to work in domestic area. HONDA therefore developed smaller humanoid "ASIMO" shown in fig. 4 than P2 and P3 developed before. Researcher in HONDA mainly concentrated themselves on achieving biped locomotion, recently vision system and voice recognition system are also installed. Development of practical arms and high-level intelligence are next issue. The conventional robot has been developed to work specific task but the robot for the next generation should have multiple functions for flexibility and then its structure and information process become complicate. We therefore think the system integration technology as well as fundamental technology is key to realize an anthropomorphic robot.
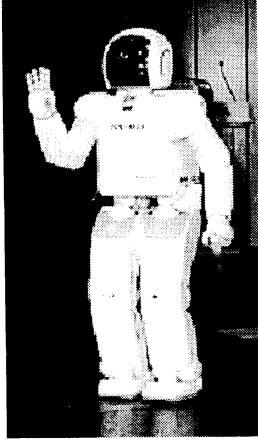
Figure 4: Humanoid robot "ASIMO" from HONDA
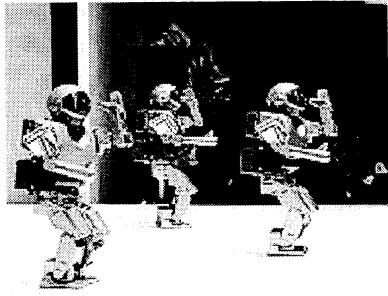
Figure 5: Humanoid robot "SDR–3X" from SONY
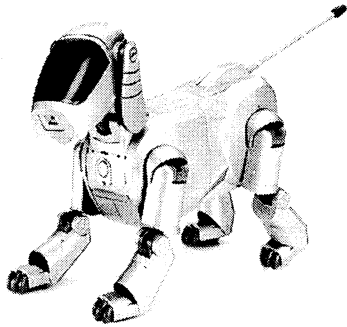
Figure 6: Pet robot "AIBO" from SONY
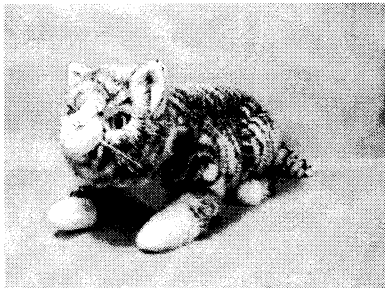
Figure 7: Pet robot "TAMA" from OMRON

	Developer	Identification	Mobility	Communication
Humanoid (ASIMO)	HONDA	–	Biped Walk	△ (Remote Control)
Humanoid (SDR-3X)	SONY	△ Vision Sensor (Only Tracking)	Biped Walk	–
Dog (AIBO)	SONY	△ Vision Sensor (Only Tracking)	Tetrapod Walk	–
Cat ("TAMA")	OMRON	–	–	–
R100	NEC	Vision Sensor (Face)	Wheel	○
Cat ("Tama")	Panasonic	–	–	○
Seal	Sankyo Aluminum Co., Ltd	–	–	–
Bird ("Ga-go")	C.A.I.	Vision Sensor (Face)	–	○
Fish	Mitsubishi Heavy Industry	–	Swim	–

Figure 8: Comparison of personal robots

3.2 Brachiation Robot

The brachiation robot is a mobile robot, which dynamically moves from branch to branch like a gibbon, namely long-armed ape, swinging its body like a pendulum [16],[17](Fig. 9). A lot of research about a brachiation type locomotion robot had been carried out. Saito et al [18] proposed the heuristic learning method for generating feasible trajectory for two-link brachiation robot. Fukuda et al [19] propose the self-scaling reinforcement learning algorithm to generate feasible trajectory with robust property against some disturbances. The reinforcement learning method builds a fuzzy logic controller with four inputs and one output. In these studies, the controller is acquired in a try-and-error learning process and a dynamics model of the two-link brachiation robot is not used for controller design. On the other hand, Nakanishi et al [22] took another approach, using target dynamics, for controlling an underactuated system. The two-link brachiation robot is an underactuated system with two degrees of freedom and one actuator. As a two-dimensional extended model, seven-link brachiation robot is studied by Hasegawa et al [21]. The seven-link brachiation robot has redundancy to locomote so that it is able to take a dexterous motion like a real ape in plane, however a dynamical locomotion robot with multi-degree of freedoms is difficult to be controlled. A hierarchical behavior architecture is adopted to design the controller with multi-input and multi-output efficiently. The behavior controllers and their coordinators in the hierarchical structure are generated using reinforcement learning method. The concept of the hierarchical behavior controller is based on the behavior-based control, which has an advantage of designing the controller for a higher-level behavior of the complex system from simpler behaviors in reasonable process.

We developed 13-link brachiation robot shown in fig.10, that has almost same dimensions and weight as a real long-armed ape. The hierarchical behavior controller shown in fig.11 generates dynamical motion con-

trolling 14 actuators.
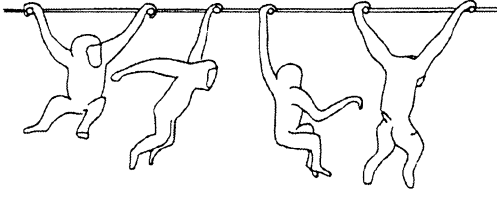
Figure 9: Brachiation motion of a long-armed ape
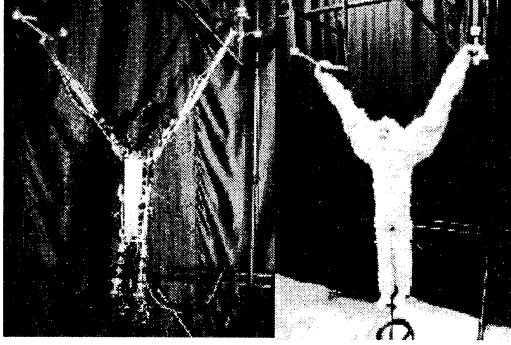
Figure 10: 13-link brachiation robot

3.2.1 Hierarchical Behavior-based Controller:
Utilization of the hierarchical behavior controller approach makes the controller designing process easier and shorter. There are some problems in this approach. One is how to adjust the behavior coordinator when the objective behavior or robot parameters are changed. We use Newton Raphson method to adjust the behavior coordinator against some environmental or task changes. This method measures the effects of the local behavior controllers to the global behavior, and changes the coefficients for them in fewer trials.

The hierarchical behavior controller for brachiation robot is designed based on behavior-based approach since the robot has multiple degrees of freedom and the objective task is complex. At first, the brachiation behavior is divided into two actions: a swing action that stores the sufficient energy prior to the transfer (preliminary swing mode), and a locomotion action that is actual transfer operation (locomotion mode). After that, these actions are decomposed into the local behaviors; leg swing, body rotation I, leg stretch, body rotation II, body lifts and arm reaching. The hierarchical behavior controller is shown in Fig.11.

3.2.2 Rescalling the Desired Trajectories from Behavior Controller:
The behavior controllers except the arm reaching behavior controller are feedforward controllers which generate the desired trajectories expressed by the cubic spline function to the feedback controllers. This desired trajectory for the actuators is rescaled by a corresponding coefficient from the behavior coordinator on the upper layer as follows.

$$yd_i(t) = r_k(y_k(t) - b_k(t)) \qquad (1)$$

where yd_i is desired trajectory to actuator i, r_k is coefficient for behavior controller k, $y_k(t)$ is actuator trajectory from behavior controller k, and $b_k(t)$ is base line connecting an initial point with an end point as follows,

$$b_k(t) = b(0)(t^* - t)/t^* + b(t^*)t/t^* \qquad (2)$$

where t^* is finishing time of the behavior.

If multiple behavior coordinators indicate coefficients to one behavior controller, summation of these values becomes new one as follows,

$$r_k = \prod_{i \in I} r_i \qquad (3)$$

where I is group that indicates coefficient to the behavior controller k.

The feedback controller makes the corresponding actuator trace the desired trajectory from behavior controller.

3.2.3 Adaptation Algorithm:
The brachiation robot is controlled by the hierarchical behavior controller explained above. To adjust the total behavior against the small changes of robot parameters or change of a desired behavior, we have three parts to be updated; the fundamental behavior controller, the behavior coordinator and both of them. The easiest part is to adjust only behavior coordinator, because of the small searching space. If the structure of the robot system or the desired task is completely changed, both the behavior controller and behavior coordinator should be rearranged. The proposed method is concerning about adaptation of only the behavior coordinator. In order to adjust the behavior coordinator, we should evaluate the effect of each behavior controller to the global behavior. Therefore, in the proposed algorithm, the relations between the each behavior and the global behavior through several trials are approximately measured and then the coefficients to the behavior controller are updated based on the relations.

The relation between change of the activation coefficient and the resultant behavior is strongly nonlinear. We however assume that the relations could be express as the multiplier of the degree of contributions and the coefficients only in a limited neighborhood of the current state as follows,

$$p(s) = W(s) \cdot r(s) \qquad (4)$$

where $p(s)$ is performance vector at step s, $W(s)$ is a gradient matrix and $r(s)$ is an activation coefficient.

The performance vector is a column of performance indices which evaluate the resultant behavior. The target activation coefficient, is derived from eq. (7).

$$\begin{aligned} e(s) &= p^* - p(s) & (5) \\ &= W(s) \cdot r^*(s) - W(s) \cdot r(s) & (6) \\ &= W(s) \cdot (r^*(s) - r(s)) & (7) \end{aligned}$$
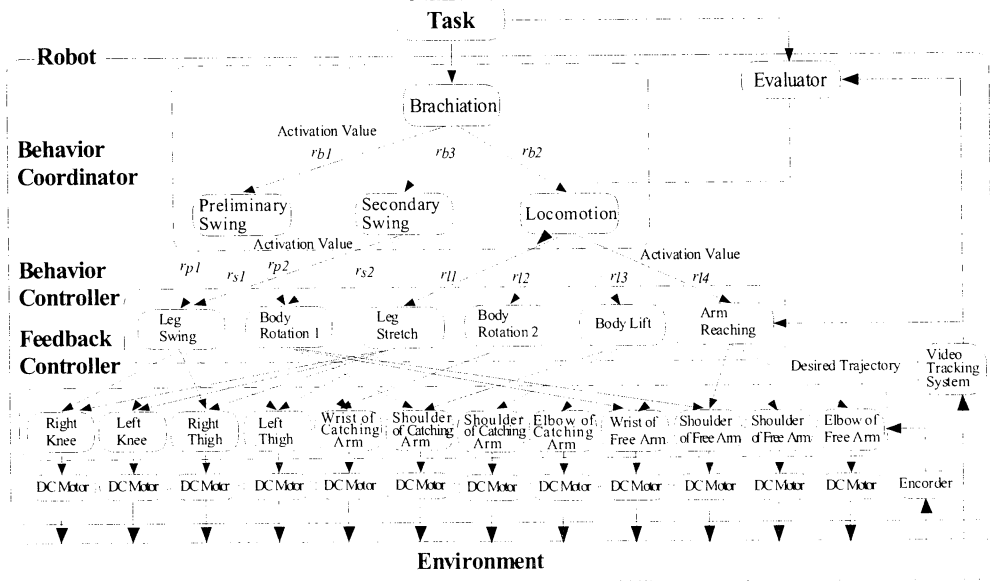
Figure 11: Behavior-based controller for brachiation robot

where $e(s)$ is error vector from the desired performance, p^* is a target performance vector and p is a current performance vector.

This calculated activation coefficient r^* is not the desired ones, because the linearized equation is adopted in the non-linear system. Therefore the target activation coefficients are searched iteratively as follows,

1. At first, evaluate its performance by making trial with activation coefficients r.

2. Explore the performance around neighborhoods area, r', r'', and r^{n-1}. These r should be linear independence each other.

3. Update gradient matrix using eq. (11), and calculate new activation coefficients using eqs. (12) and (12).

4. Check its performance with the new activation coefficients. If the behavior is not insufficient, go to step 2.

$$R(s) = (r(s), r'(s), r''(s), \ldots, r^{n-1}(s))(8)$$
$$P(s) = (p(s), p'(s), p''(s), \ldots, p^{n-1}(s))(9)$$
$$W(s) = P(s) \cdot R(s)^{-1} \tag{10}$$
$$\Delta r(s) = W(s)^{-1} \cdot e(s) \tag{11}$$
$$r(s+1) = r(s) + \Delta r(s) \tag{12}$$

3.2.4 Experiments: We applied the proposed method to two cases. One is the case when the objective task is changed. The other is the case when the new behavior should be generated based on the primitive behavior.

This adaptation algorithm adjusts four coefficients from behavior coordinator "locomotion" to four behavior controllers when the branch intervals are extended from 90cm to 101cm. The feasible coefficients are obtained in four steps and the figure 12 shows the trajectories of tip of the free hand both before and after adaptations.
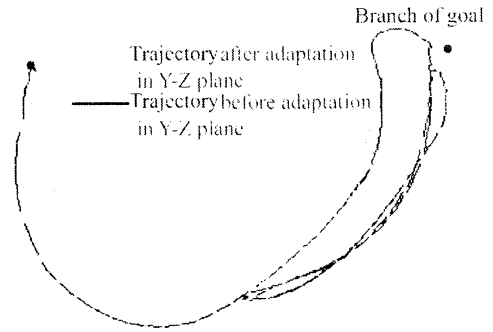
Figure 12: Free hand's trajectories before and after adaptation

In order to achieve the continuous locomotion, the end posture of first swing should be important for the second swing. We therefore applied the adaptation algorithm to tune the secondary swing motion controller with two parameters. Figure 13 is a Stroboscopic photo showing the continuous motion with $33msec$ interval.

4 Summary

This paper presented some recent researches in the field of intelligent robotic system. The computational
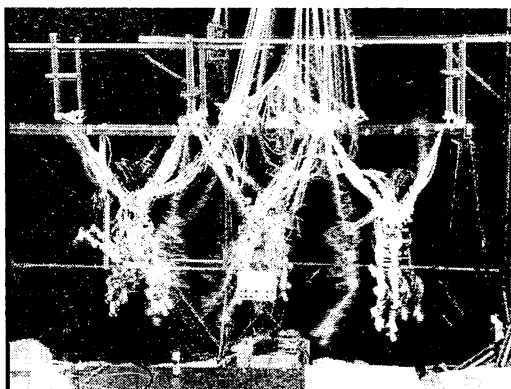
Figure 13: Continuous locomotion

intelligence means capability of learning, adaptation and evolution using neural network, fuzzy logic and evolutionary computation. The synthesis of neural network, fuzzy logic and evolutionary computation will be realize advanced information processing and hierarchical structure organization. After that, this paper showed the pioneer robots for personal and domestic use announced by industrial companies in Japan. They required more intelligent capability for perception and decision making algorithm from ambiguous information and stored knowledge as well as system integration technology to realize a sophisticated robot. Finally we showed a brachiation robot like a long-armed ape as an example of adaptation algorithm with hierarchical behavior-based control architecture.

References

[1] S.J.Russell, P.Norvig, Artificial Intelligence, Prentice-Hall, Inc. (1995).

[2] J.A.Anderson, E.Rosenfeld, Neurocomputing - Foundations of Research, The MIT Press (1988).

[3] M.Palaniswami, Y.Attikiouzel, R.J.Marks II, D.Fogel, T.Fukuda, Computational Intelligence - A Dynamic System Perspective, IEEE Press (1995).

[4] L.A.Zadeh, Fuzzy Sets, Information and Control, Vol.8, 338-353 (1965).

[5] J.-S.R.Jang, C.-T.Sun, E.Mizutani, Neuro-Fuzzy and Soft Computing, New Jersey: Prentice-Hall, Inc. (1997).

[6] C.G.Langton, Artificial Life -An Overview, The MIT Press (1995).

[7] R.J.Marks II, Intelligence: Computational Versus Artificial, IEEE Trans. on Neural Networks, Vol.4, No.5, 737-739 (1993).

[8] D.E.Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Welsey (1989)

[9] D.B.Fogel, Evolutionary Computation, IEEE Press (1995).

[10] J.Koza, Genetic Programming, The Mit Press (1992).

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

[12] R.Brooks, A Robust Layered Control System for a Mobile Robot, IEEE Journal of Robotics and Automation, RA-2-1, 14-23 (1986).

[13] Marco Colombetti, Marco Dorigo, Giuseppe Borghi, Behavior Analysis and Training - A methodology for Behavior Engineering, IEEE Transaction on Systems, Man, And Cybernetics, Part B: Cybernetics, Vol.26, No.3, 365-380 (1996).

[14] Jun Tani, Model-Based Learning for Mobile Robot Navigation from the Dynamical Systems Perspective, IEEE Transaction on Systems, Man, And Cybernetics, Part B: Cybernetics, Vol.26, No.3, 421-436 (1996).

[15] T.Fukuda, N.Kubota, Intelligent Robotic System -Adaptation, Learning and Evolution, Proc. of The Third International Symposium on Artificial Life and Robotics 40-45 (1998).

[16] T. Fukuda, H. Hosokai and Y. Kondo, Brachiation Type of Mobile Robot, Proc. IEEE Int. Conf. Advanced Robotics, pp. 915-920,(1991).

[17] Spong, M. W., Swing Up Control of the Acrobot, Proc. 1994 IEEE International Conference on Robotics and Automation, pp. 2356-2361, (1994)

[18] T. Fukuda, F. Saito and F. Arai, A Study on the Brachiation Type of Mobile Robot(Heuristic Creation of Driving Input and Control Using CMAC), Proc. IEEE/RSJ Int. Workshop on Intelligent Robots and Systems, pp. 478-483, (1991).

[19] T. Fukuda Y. Hasegawa K. Shimojima and F. Saito, Self Scaling Reinforcement Learning for Fuzzy Logic Controller, IEEE International Conference on Evolutionary Computation, pp. 247-252, Japan, (1996)

[20] J. Nakanishi, T. Fukuda and D.E. Koditschek, Experimental implementation of a Target dynamics controller on a two-link brachiating robot, IEEE International Conference on Robotics and Automation, pp. 787-792, (1998)

[21] Y. Hasegawa and T. Fukuda, Learning Method for Hierarchical Behavior Controller, IEEE International Conference on Robotics and Automation, pp.2799-2804, (1999)

[22] Y. Hasegawa, Y. Ito and T. Fukuda, Behavior Coordination and its Modification on Brachiation-type Mobile Robot, 2000 IEEE International Conference on Robotics and Automation, San Francisco, pp3984-3989, (2000)

Intelligent Artefacts

Henrik Hautop Lund

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

hhl@mip.sdu.dk
www.adaptronics.dk

Abstract

Based on principles from modern artificial intelligence and artificial life, I developed the concept of 'programming by building' using intelligent artefacts. Miniaturisation of computing makes it possible to employ new educational practises and revolutionise our idea about programming through the use of technological components. With the development of intelligent building blocks it becomes possible to 'program by building'. The construction with intelligent building blocks results not only in the development of a physical structure, but also in the development of a functionality of the physical structure. So construction of functionality can happen with physical building blocks that each contains computational processing and communication. The intelligent building blocks are the support for the concept of programming by building and have different forms and material properties.

Introduction

Miniaturisation of computing makes it possible to employ new educational practises and revolutionise our idea about programming through the use of technological components. With the development of intelligent building blocks it becomes possible to 'program by building'. The construction with intelligent building blocks results not only in the development of a physical structure, but also in the development of a functionality of the physical structure. So construction of functionality can happen with physical building blocks that each contains computational processing and communication. The intelligent building blocks are the support for the concept of programming by building and have different forms and material properties.

The reason for moving towards physical programming of functionality (rather than traditional programming on a personal computer) is inspired by Piaget's [1,2] and Vygotsky's [3] work in the field of educational psychology. Piaget suggested and showed that hands-on

experimentation is the essential basis for cognitive development, and Vygotsky viewed knowledge as a process, which basically depends on technological and cultural scaffolding [3].

Looking at the Piaget's definition of stages, it becomes clear that interaction might play a major role in children learning, and that the younger children are the more important hands-on experiences would be.
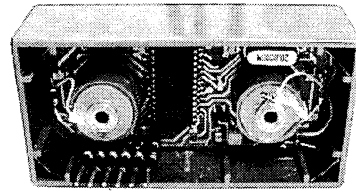
Figure 1. A building block with micro processor and communication channels. Copyright 2002, H. H. Lund.

Also, classical constructionism and guided constructionism [4] with its roots in the work by J. Piaget suggest that the best way to learn about an artefact is to actually build the artefact: "Children have real understanding only of that which they invent themselves, and each time that we try to teach them something too quickly, we keep them from reinventing it themselves."

Further, research in the field of new artificial intelligence, also called embodied artificial intelligence [5] suggests that the embodiment and situatedness play a major role in the development of intelligence. Therefore, also in the scientific process of understanding intelligence it becomes important to move towards physical systems that allow the control to be embodied and situated (e.g. research show that robot bodies and brains should be co-evolved [6]).

What may look as a conflict between much modern research on developing *autonomous* systems, and the educational research putting emphasis on *interaction*

turns out to be the corner stone for developing 'programming by building'.

Intelligent Building Blocks

In order to exemplify the concept of 'programming by building', a series of intelligent building blocks were developed. Each intelligent building block contains processing power and communication capability. When two building blocks are physically connected they can communicate to each other and process the received information from neighbours. The building blocks can take various forms and be implemented in various materials, but for simplicity and for better visualisation of the concept, let us take the example of housing them in LEGO DUPLO bricks[1].

Each brick is equipped with an electronic circuit containing a PIC16F876 40-pin 8 bit CMOS Flash microcontroller and a number of serial two-way connections. In the case of rectangular LEGO DUPLO, each brick contains four serial two-way connections, two connections on the top and two connections on the bottom of each building block. In other implementations, there may be more or less connections, for instance there may be six connections (one on each side) in a cubic building block.
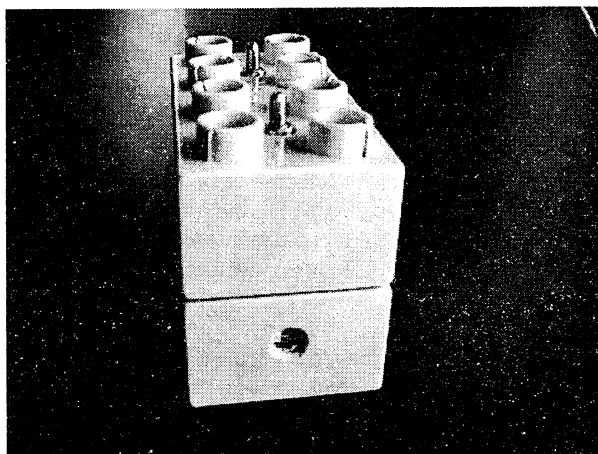
Figure 2. In the implementation in LEGO DUPLO bricks, the intelligent building blocks may have two serial connections on the top (in the centre) and on the bottom. Further, power can be transferred through connectors, as seen on the sides of each stud on the top of the intelligent brick. In this case, a standard building block is placed on top of a LDR sensor building block. Copyright 2002, H. H. Lund.

---

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

When two building blocks are physically connected together, they may communicate with each other over the serial connection(s). In the sending building block, signals are sent to one of the connections and these signals are received by the attached connection in the receiving building block. In a typical set up, each building block will receive input on its communication channels, process this input, and then send the output of the processing procedure as output to the communication channels. A construction of such building blocks will have functionality defined by the physical construction (i.e. the topology), the input, the processing procedure in the individual building blocks, and the communication scheme. If input, processing, and communication are pre-defined, the user of such a system can decide the functionality of the system exclusively by manipulating the physical structure. However, in extensions of such a system, it may be desirable to allow input to be decided at run-time, for instance through the inclusion of sensors.

A series of such building blocks were designed. All include the standard functionality of being able to process and communicate, but some include additional hardware, so amongst others, there exist input and output building blocks as indicated in table 1. Further, there exists a rechargeable battery building block and a battery building block for standard 9V batteries. Also, some standard building blocks contain small back-up batteries that allow a construction to function for a period of time (up to 5 minutes if no actuator building blocks are included) when detached from the centralised battery building block. It is possible to construct a number of other building blocks not included in the list of currently available building blocks, e.g. building blocks with digital compass, sonar, accelerometer, etc.
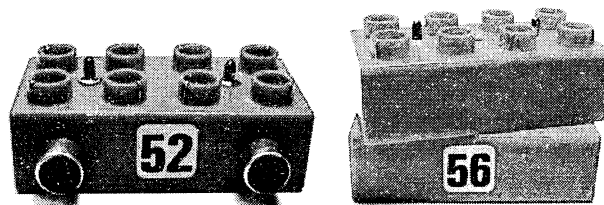
Figure 3. An example of intelligent building blocks implemented in LEGO DUPLO. Left: sensor building block that contains two microphones. Right: motor building block that contains a servo motor that allows the top element to turn. Copyright 2002, H. H. Lund.

Table 1. Classification of different hardware types of building blocks and their corresponding, possible functionality.

General type	Specific type	Functionality
Standard building blocks	Standard	Processing and communication.
	Back-up battery	As standard, but functions with own battery.
Input building blocks	LDR sensor	Light level.
	Two LDR sensors	Light level, light direction.
	Two microphones	Sound level, direction of sound.
	IR sensor	Infra red light level and patterns.
	Touch sensor	Pressed or not.
	Potentiometer	Turn level.
Output building blocks	Double motor brick	Turn +/- 45 degrees.
	Motor brick	Axe turn 360 degrees.
	Eight LEDs	Light in patterns.
	Sound generator	Sounds.
	IR emitter	Wireless communication.
	Display	Displays messages.

With these building blocks, it is possible to construct a huge variety of physical objects with various functionalities. The processing in the physical construction is distributed among all the building blocks, and there is no central control opposed to traditional computerised systems (e.g. traditional robots). The distribution of control is obtained by allowing processing within each individual building block.

The importance of distributed control as opposed to central control is illustrated by Brooks' [7] seminal work on behaviour-based control. A paradigm shift was based on Brooks work showing how to facilitate the development of control based on a task decomposition into parallel behaviours rather than into functional units in a central control. The paradigm shift had major impact on and was reinforced by other research fields, so that research on multi-agent systems, ant systems, networks

(connection machine, Beowulf clusters, etc.), neural networks, complex systems and emergence put emphasis on distribution of control. Further, the current trend towards distributed control in artificial intelligence can be illustrated by much research moving away from traditional landmark problems such as computer chess that facilitate solutions based upon central control due to complete, central information about the environment (the chessboard and the chess pieces).

However, as illustrated by the vast number of research fields focusing on distributed control, a major challenge is found in the definition of processing and coordination in a distributed system. Here, our work exploits distributed processing in the single building blocks. This differs from other research directions that try to co-evolve control and morphology, e.g. [8], where the building blocks do not contain processing power. In such related work, central control is promoted, and a number of physical shapes are promoted as possible building blocks. We do not agree that these choices are necessarily correct.

Experiments

We implemented both arithmetic building blocks, behaviour building blocks and neural building blocks, and other control methods can easily be implemented in intelligent building blocks. Arithmetic building blocks contain arithmetic operations such as addition, subtraction, division, multiplication, and children can therefore perform mathematical exercises by building physical structures with the intelligent building blocks. The behaviour building blocks contain primitive behaviours that can be executed according to a schedule in the building blocks. For instance, they may be executed in a sequential manner according to their position in the physical construction, or extensions may lead to a behaviour-based system with parallel execution of primitive behaviours and arbitration in the actuator building blocks. In the following, however, I will exemplify the neural building blocks.

Neural building blocks are implementation of artificial neural networks in the intelligent building blocks. Each building block models a neuron with soma, axons and dendrites. The soma is modelled with the processing in the microcontroller, the axons and dendrites with input and output channels. By attaching neural building blocks (neurons) together, it is possible to create an artificial neural network. The physical structure of the collection of neural building blocks decides the topology of the

artificial neural network. As in other types of control with the intelligent building blocks, sensor building blocks can function as input and actuator building blocks can be controlled as output. Therefore, it becomes possible to create artefacts with sensors and actuators controlled by the distributed control in the artificial neural network.

As an example of how to use the intelligent artefacts, let us look at the task of calculation the results of arithmetic expressions. Here, it may be natural to use building blocks that are implemented to perform arithmetic operations. So this kind of building blocks will be used here (though we can of course imagine using other implementations). The task is 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 task has been solved correctly for a solution that presents the right result with regards to every possible input.

Figure 4. A construction with arithmetic bricks that performs an arithmetic calculation: (x+y)*z. The user can set the value of the small bricks 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 brick. Copyright 2002, H. H. Lund.

This system with arithmetic bricks solves this task quite easily as shown in figure 4, because this system is built for arithmetic operation. The correct result will only be presented on the display (the brick 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 figure 4, where the binary display brick 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 brick).

Conclusion

It is possible to make intelligent artefacts that, in some cases, allow the user to 'program by building' a physical structure, as shown with the arithmetic expression example. We performed numerous other tests with behavior bricks and neuron bricks. Space limitations does not allow presentation of these results here, so they will be presented in future work.

Acknowledgements

References

[1] J. Piaget. La construction du réel chez l'enfant. Delachaux & Niestlé, Paris.
[2] J. Piaget. The Science of Education and the Psychology of the Child. NY: Grossman, 1970.
[3] L. S. Vygotsky. *Thought and language*. Cambridge; MIT Press, 1986.
[4] H. H. Lund. Robot Soccer in Education. Advanced Robotics Journal,13:8, 737-752, 1999.
[5] R. Pfeifer and C. Scheier. Understanding Intelligence. MIT Press, Cambridge, MA, 1999.
[6] H. H. Lund. Co-evolving Control and Morphology with LEGO Robots. In Hara and Pfeifer (eds.) Morpho-functional Machines, Springer-Verlag, Heidelberg, 2002.
[7] R. A. Brooks. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, 2(1):14--23, 1986.
[8] H. Lipson and J. Pollack. Automatic Design and Manufacture of Artificial Lifeforms. Nature 406, 974-978, 2000.

The Waves of Life

John L. Casti
Santa Fe Institute
1399 Hyde Park Road
Santa Fe, NM 87501, USA

Abstract

Elliott wave theory has been used for many years as a tool for analyzing price movements in financial markets. More recently, attention has been directed toward whether the Elliott wave principle might apply in a much broader context than simply in finance. In particular, whether the same wave-like structures underlie all social movements. This paper presents a brief overview of the Elliott wave principle, followed by an examination of how the principle works for social and behavioral phenomena.

1 Accounting for Things

Ralph N. Elliott was a Los Angeles accountant who found himself in frail health and unable to find work during the Depression. While recuperating, he had plenty of free time to investigate why the market had lost 90 percent of its value over a three-year period. He became convinced that there is a repetitive pattern that market indexes like the Dow Jones Industrial Average (DJIA) go through, and that an understanding of these cycles would enable the alert investor to foresee bull markets like the 1920s, bear markets like 1966–1974 and even great crashes such as those of October 1929 and October 1987. Of course, to say that the Dow moves in cycles is not really saying much. What we need to know is the *kind* of cycle that the market goes through. And it is just this sort of information that the Wave Principle purports to provide.

Already a devoted reader of Robert Rhea's *Dow Theory Letter,* which was the technician's icon of the era, and as a hands-on specialist in corporate financial rescue, Elliott was no stranger to market analysis, or the "ebbs" and "flows" of the business cycle. After a few years of painstaking research, the missing pieces of a very fascinating puzzle began to fit together. Elliott observed, "Human emotions are rhythmical; they move in waves of a definite number and direction. The phenomenon occurs in all human activities, be it business, politics, or the pursuit of pleasure. It is particularly evident in those free markets where public participation in price movements is extensive." The basic idea of what Elliott discovered is rather simple to describe.

The fundamental pattern upon which the entire Elliott edifice rests is shown in Figure 1, which displays a sequence of up and down price movements constituting a complete cycle of eight waves. Typically, these waves are the ups and downs of a market indicator like the DJIA or the FTSE 100 Index on the London Exchange. Waves 1, 3, and 5 are called *impulse* waves, while waves 2 and 4 are termed *corrective* waves. So one complete Elliott cycle consists of eight waves divided into two distinct phases: Numbered phases are in the direction of the main trend, while lettered phases move against the trend.
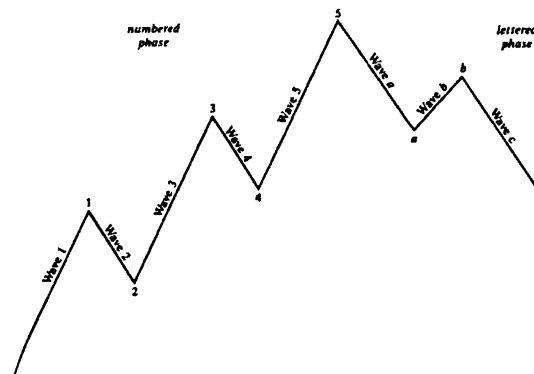
Figure 1.

Elliott found that following completion of the above cycle, a similar but higher-level cycle begins: Another five-wave up pattern followed by another down pattern of three waves correcting the up pattern. The overall situation is shown in Figure 2, while Figure 3 carries the idea through one entire market cycle. Figure 2 shows the crucial fact that each of the numbered and lettered phases is actually a wave itself, but of one

degree higher than its component waves. The numbers shown in Figure 3 for the bull and bear cycles are the number of "*cycle*" waves, "*primary*" waves, "*intermediate*" waves, and so on present in the overall movement.
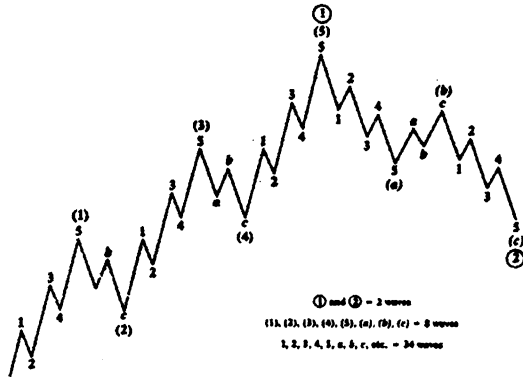
Figure 2.
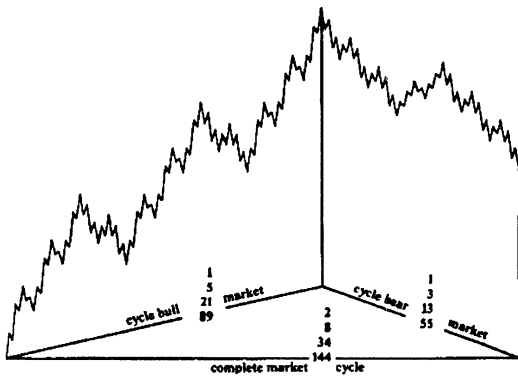
Figure 3.

The distilled essence of Elliott-wave formation consists of the following four principles:

- Action is followed by reaction.

- Impulse waves subdivide into five waves of lower degree, while corrective waves subdivide into three waves of lower degree.

- A complete cycle consists of an eight-wave movement (five up and three down), which then becomes two subdivisions of the wave of next higher degree.

- The time frame does not enter into the pattern, so that the waves may be stretched or compressed along either the horizontal or the vertical axis without losing the underlying pattern.

How many cycles, subcycles, and sub-subcycles are there? The Wave Principle is hierarchical in that each wave has component waves and is itself a component of a larger wave. Wave degrees, or relative sizes, are denoted by names such as primary, intermediate and minor. Once you name any wave by degree, all other degrees take on their roles as smaller or larger interconnected structures. Elliott named nine degrees of waves, from those lasting centuries to those lasting mere hours. The actual number of degrees may be infinite since the patterns show up even on one-minute graphs of aggregate stock prices and are likewise presumed to expand indefinitely into larger and larger degrees.

2 The Ups and Downs of Finance

Certainly the area in which the Wave Principle has been most extensively applied is in finance. The diagram in Figure 4 shows how the Principle works with real data from the New York Stock Exchange. The various waves and subwaves are marked for the DJIA over the period 1932–2002.
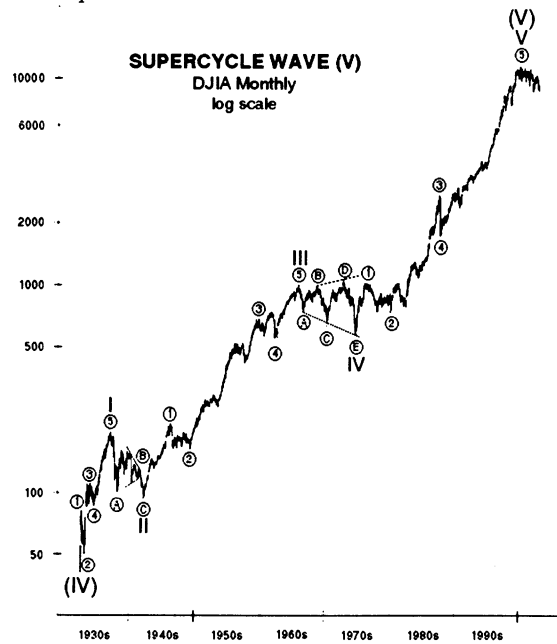
Figure 4.

It's often been said that price movements in financial markets simply reflect the collective beliefs of investors about the future. If the majority are optimistic, prices rise; if not, they fall. But this psychology

of belief is not confined just to financial markets. It pervades all aspects of human interaction. So it should come as no surprise that the Elliott Wave Principle governs all human social interaction.

3 Herding Behavior and the Forces of History

Psychologists and neurophysiologists have shown time and again that there are strong biological mechanisms in the human brain that give rise to what's called "herding behavior": the tendency for humans to react impulsively to situations as individuals, which leads to collective behavior displaying a kind of intelligence, or at least structured pattern. Crashes and bubbles in stock market indices like the DJIA are excellent examples, since the individual investors collectively tend to "herd" in their buying or selling actions.

Historians, too, note this type of behavior in the lack of vocal critics in countries taken over by fascists, communists, witch-burners and other repressive regimes. This type of "going along with the crowd" mentality has strong evolutionary explanations in terms of simple survival value, the basic principle being that if you have too little basis upon which to make a judgment, the only alternative is to assume that the herd knows where it's going. Financial guru Robert Prechter argues forcefully for the applicability of Elliott wave patterns as an organizing principle for many types of social behavior outside the realm of finance.

Prechter's basic idea is that the units in a social system—investors, voters, music fans, shoppers—tend to make their individual decisions on the basis of what they see others doing. In other words, they herd. These decisions are then translated into a "social mood", which is measured by things like the DJIA, hemlines levels in women's skirts, types of musical lyrics, or attendance levels at baseball games. What sets Prechter's "socionomics" apart from conventional wisdom is that he completely turns around the idea of cause-and-effect in social systems.

If social moods have a definite pattern, they certainly cannot be the result of random events. And there is not a shred of evidence to suggest that a social mood is the result of events that are themselves structured to produce the Elliott wave patterns they display. Thus, the only possible direction of causality is to assume that events do not shape social mood; rather, social mood shapes events. Thus, collective mood shapes the nature of social interaction, and con-

sequently the resulting social actions and events. So, for instance, production, recession, political news and even major conditions such as war and peace to not buffet the social mood; rather, the trends in social mood induce people to take actions that show up as economics, politics and all other manner of social trends that make history.

4 Socionomics and the Enron Scandal

The recent Enron scandal in the United States is a good example to illustrate socionomic thinking. Newspapers and magazines trumpeted the conventional direction of causality for weeks: The Enron scandal deeply discouraged investors. Socionomic thinkers argue just the opposite: Discouraged investors precipitated the Enron scandal. The argument supporting this socionomic line of reasoning is that investors, in general, knew nothing about Enron's malpractices prior to or anytime during the stock market's decline. Moreover, throughout the drama of the scandal, which unfolded primarily from November to March, the market actually *advanced*. So it is abundantly clear that as the Enron scandal developed, investor and consumer psychology improved and stock prices rose. Thus, it is completely false that the Enron scandal discouraged investors.

Looking at the performance of the market throughout 2001, we can see the alternative view—socionomic causality—at work in the chronology of events. Stock prices were falling for a full eighteen months prior to the scandal. Enron stock also retreated, thus taking away support for financing its deals. By the time that negative mood trend brought the Enron scandal to light, the trend toward an increasingly negative mood was already over. In fact, by late September 2001, it was time for the market to make its largest move upward in over a year-and-a-half. So corporate misdeeds did not crush stock prices; crashing stock prices eventually drew attention to the corporate misdeeds. And what caused these misdeeds to expand so greatly to begin with? Nothing other than the mass psychology of the stock mania. It was the psychological climate of the bull market that encouraged companies to mislead investors and which prompted investors to accept all manner of corporate behavior that reflected and supported their ebullient mood. That's socionomic thinking for you.

It's amusing to note that a financial guru from the 1980s like Prechter has used socionomic principles to anticipate the peak in the popularity of a public person—like himself. Using the number of subscribers

to his publication, the *Elliott Wave Theorist,* as a measure of popularity, he saw that the subscription levels obeyed an EW pattern of their own. As wave 5 of the advance in the pattern began to slow in late 1987, he knew that the end of his ride as a guru was near. And sure enough, despite the fact that 1988 was one of his best forecasting years ever, some members of the media had had enough of Prechter and began to attack the persona that they and their colleagues had overpromoted.

A similar analysis using the EW pattern for Michael Jackson suggested the pop icon's fall from grace. Within a year of his signing a massive recording contract with SONY on March 19, 1991, the singer's image began slipping. That was the A wave downward. Then came a B wave bounce when Jackson performed at the Super Bowl halftime in January 1993. By the end of that year, though, his image was collapsing in an astoundingly powerful C wave—accusations of child molestation, cancellation of a world tour, termination of major advertising contracts, lawsuits for millions of dollars from songwriters claiming plagiarism, withdrawal of a prestigious award due to be given him by a major university—all part of an often-predictable EW pattern.

While the Wave Principle is good for recognizing tops, it is perhaps even more valuable in giving perspective on turns for the better in times of crisis. R. N. Elliott used his waves to announce, in the middle of the worst of World War II, that a multi-decade stock-market advance was about to begin. Prechter did the same in the midst of recession in September 1982 by announcing that a "super bull market" had begun and graphing a forecast for a five-times multiple in stock values.

As a final point, it's tantalizing to speculate that if society's actions unfold over time spans of ten minutes, ten days, ten weeks, ten months, ten years and ten decades, why not ten centuries or ten millennia? Because these waves manifest themselves as events, the EW theory can be used to probe social events into the deep past. For instance, looking at trends in Western culture since Roman times, we can rely on market price data from the year 1690 onward to correlate with social trends. But EW theory then enables us to construct waves prior to this time on the basis of social events and trends. Thus, in a real sense, the EW theory can be thought of as a kind of "time machine" for going where no theory of social change has gone before.

References

[1] Frost, A. and R. Prechter. *Elliott Wave Principle.* New Classics Library, Gainesville, GA. 1978-1998.

[2] Prechter, R. *The Wave Principle of Human Social Behavior.* New Classics Library, Gainesville, GA, 1999.

[3] Casti, J. *Searching for Certainty.* Abacus Books, London, 1992.

Application of uncertain variables and learning algorithm to task allocation in multiprocessor systems*

Z. Bubnicki

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

Keywords: uncertain systems, learning algorithms, knowledge-based systems, intelligent systems, multiprocessor systems

Abstract

The paper is concerned with a task allocation in a set of parallel processors described by knowledge representations in a form of relations with uncertain execution times. In the first part the unknown parameters are assumed to be values of so called uncertain variables described by certainty distributions given by an expert. It is shown how to determine the allocation maximizing the certainty index that a requirement concerning the execution time is satisfied. In the second part learning algorithms consisting in step by step knowledge validation and updating are presented.

1. Introduction

In recent years two new concepts have been developed for the decision making in a class of uncertain systems described by a relational knowledge representation with unknown parameters:
1. The uncertain variables described by an expert and characterizing his / her knowledge concerning approximate values of the unknown parameters [1, 2, 3, 4].
2. The learning algorithms consisting in *step by step* knowledge validation and updating based on the results of current observations [5].

The purpose of this paper is to show how the uncertain variables and the learning algorithms may be applied to the allocation of computational tasks in a set of parallel processors. We shall consider a rather simple allocation problem consisting in the distribution of a great number of elementary tasks (elementary programs or parts of programs).

The uncertain variable $\bar{x}$ is defined by a set of variables X and a *certainty distribution* $h(x) = v(\bar{x} \cong x)$ where $v \in [0, 1]$ denotes the certainty index of the soft property: "$\bar{x}$ is approximately equal to x". The soft property $\bar{x} \tilde{\in} D_x$ means: "$\bar{x}$ approximately belongs to D_x" or "an approximate value of $\bar{x}$ belongs to D_x", where D_x

is a subset of X. The certainty index of $\bar{x} \tilde{\in} D_x$ is defined as follow:

$$v(\bar{x} \tilde{\in} D_x) = \max_{x \in D_x} h(x).$$

In Sec. 2 we assume that the unknown execution times of the tasks are described by their certainty distributions given by an expert and we show how to determine the allocation for which the requirement concerning the execution time of the global task is satisfied with the greatest certainty index. Section 3 presents the algorithms of *step by step* modification of the allocation in the successive cycles with the same global task.

2. Application of uncertain variables to task allocation in a set of parallel processors

Assume that the global computational task to be distributed may be decomposed into N separate parts (programs or parts of programs) which may be executed simultaneously by the separate processors. Each partial task is characterized by an upper bound of the execution time τ_j for j-th processor ($j = 1, 2, ..., k$), and τ_j is assumed to be the same for each partial task. The decision problem consists in the determination of the numbers of the partial tasks $n_1, n_2, ..., n_k$ assigned to the processors taking into account the execution time $T = \max\{T_1, T_2, ..., T_k\}$ where T_j is the execution time for j-th processor; $n_1 + n_2 + ... + n_k = N$. If N is sufficiently large, we can determine the decisions $u_j \in R^+$ (any positive numbers) satisfying the constraint $u_1 + u_2 + ... + u_k = N$ and then obtain n_j by rounding off u_j to the nearest integer. Let us denote $T_j \overset{\Delta}{=} y_j$, $T \overset{\Delta}{=} y$, $\tau_j \overset{\Delta}{=} x_j$, $u = (u_1, u_2, ..., u_k) \in U$ and

$x = (x_1, x_2, ..., x_k) \in X$. Then the knowledge representation of our system is as follows

$$y_j \leq x_j u_j, \quad j = 1, 2, ..., k, \qquad (1)$$

$$y = \max\{y_1, y_2, ..., y_k\}. \qquad (2)$$

The unknown parameters x_j are assumed to be values of uncertain variables $\bar{x}_j$ described by the certainty distributions $h_{xj}(x_j)$ given by an expert. The allocation problem may be formulated as an optimization problem consisting in finding the optimal allocation u^* which maximizes the certainty index of the soft property: "the set of possible values y approximately belongs to $[0, \alpha]$" (i.e. belongs to $[0, \alpha]$ for an approximate value of $\bar{x}$).

Optimal allocation problem: For the given h_{xj} ($j \in \overline{1,k}$) and N find the allocation $u^* = (u_1^*, u_2^*, ..., u_k^*)$ maximizing the certainty index

$$v(u) = v\{D_y(u; \bar{x}) \widetilde{\subseteq} [0, \alpha]\} = v[y(u; \bar{x}) \widetilde{\leq} \alpha] \qquad (3)$$

where $D_y(u; x)$ denotes the set of possible values y for the fixed u, $y(u; x)$ denotes the value y for the fixed u and the soft property " $D_y(u; \bar{x}) \widetilde{\subseteq} [0, \alpha]$ " is denoted here by " $y(u; \bar{x}) \widetilde{\leq} \alpha$ ". From (1), (2) and (3)

$$v(u) = v\{[y_1(u_1, \bar{x}_1) \widetilde{\leq} \alpha] \wedge [y_2(u_2, \bar{x}_2) \widetilde{\leq} \alpha] \wedge ...$$
$$\wedge [y_k(u_k, \bar{x}_k) \widetilde{\leq} \alpha].$$

Then

$$u^* = \arg\max_u \min_j v_j(u_j)$$

where

$$v_j(u_j) = v[y_j(u_j, \bar{x}_j) \widetilde{\leq} \alpha] = \max_{x_j \in D_{xj}(u_j)} h_{xj}(x_j) \qquad (4)$$

and $D_{xj}(u_j)$ is described by the inequality $x_j \leq \alpha u_j^{-1}$.

It may be shown that the optimal solution u^* may be obtained from the set of equations $v_1(u_1) = v_2(u_2) = ... = v_k(u_k)$, $u_1 + ... + u_k = N$.

Example: Assume that $h_{xj}(x_j)$ has a triangular form presented in Fig. 1. Using (4) it is easy to obtain the following formula for the function $v_j(u_j)$ presenting the certainty index that in j-th processor the execution time is less than α:

$$v_j(u_j) = \begin{cases} 1 & \text{for} & u_j \leq \dfrac{\alpha}{x_j^*} \\ \dfrac{1}{d_j}(\dfrac{\alpha}{u_j} - x_j^*) + 1 & \text{for} & \dfrac{\alpha}{x_j^*} \leq u_j \leq \dfrac{\alpha}{x_j^* - d_j} \\ 0 & \text{for} & u_j \geq \dfrac{\alpha}{x_j^* - d_j} \end{cases}$$

$$(5)$$
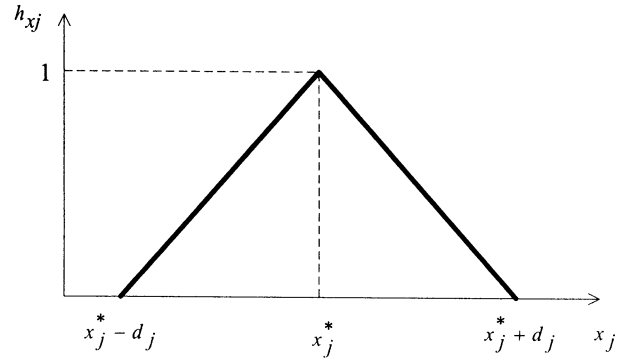
Figure 1. Certainty distribution in the example.

For two processors ($k = 2$) the decision u_1^* may be found by solving the equation $v_1(u_1) = v_2(N - u_1)$ and $u_2^* = N - u_1^*$. Using (5) we obtain the following result:

1. For

$$\alpha \leq \frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2} \qquad v(u) = 0 \text{ for any } u_1. \ (6)$$

2. For

$$\frac{N(x_1^* - d_1)(x_2^* - d_2)}{x_1^* - d_1 + x_2^* - d_2} \leq \alpha \leq \frac{Nx_1^* x_2^*}{x_1^* + x_2^*} \qquad (7)$$

u_1^* is a root of the equation

$$\frac{1}{d_1}(\frac{\alpha}{u_1} - x_1^*) = \frac{1}{d_2}(\frac{\alpha}{N - u_1} - x_2^*) \qquad (8)$$

satisfying the condition

$$\frac{\alpha}{x_1^*} \leq u_1^* \leq \frac{\alpha}{x_1^* - d_1} \quad \text{and} \quad v(u^*) = v_1(u_1^*). \quad (9)$$

3. For

$$\alpha \geq \frac{Nx_1^* x_2^*}{x_1^* + x_2^*} \qquad v(u^*) = 1 \quad \text{for any } u_1 \text{ satisfying the}$$

condition

$$N - \frac{\alpha}{x_2^*} \leq u_1 \leq \frac{\alpha}{x_1^*}. \qquad (10)$$

In the case (6) α is too small (the requirement is too strong) and it is not possible to find the allocation for which $v(u)$ is greater than 0. For example, if $U = 200$, $\alpha = 200$, $x_1^* = 2$, $x_2^* = 3$, $d_1 = d_2 = 1$ then we have the case (7). Using (8) yields $u_1^* = 125$, $u_2^* = 75$, $v(u^*) = 0.6$ what means that 125 partial tasks should be assigned to the first processor and 75 should be assigned to the second processor, and the requirement $T \leq \alpha$ will be satisfied with the certainty index 0.6.

3. Learning algorithms

Let us assume that there is no *a priori* knowledge of the unknown parameters $\bar{x}_j$ in the form of h_{xj} given by an expert but it is possible to apply learning process using the results of current observations. The learning process will consist in *step by step* estimation of $\bar{x} = (\bar{x}_1,...,\bar{x}_k)$ as a result of the knowledge validation and updating, and using the successive estimates to the current determination of the allocation. Let us denote by $u^{(n)}$ and $T^{(n)}$ the allocation $u = (u_1,...,u_k)$ and the execution time T in n-th step, respectively. The considerations will be presented for the knowledge representation in the form of inequalities

$$y_j \leq H_j(u_j, x_j) \tag{11}$$

where $y_j = T_j$ is the execution time and H_j is a known function. In particular, (11) has the form (1). For the known values x_j we may determine the greatest set $D_u(x)$ of the allocations $u = (u_1, u_2,...,u_k)$ such that the implication $u \in D_u(x) \rightarrow y \leq \alpha$ is satisfied and $D_u(x) \subseteq \bar{\Delta}_u$ where

$$\bar{\Delta}_u = \{u \in U : u_1 + ... + u_k = N; \bigwedge_{j \in \overline{1,k}} u_j \geq 0\}.$$

It is easy to note that $D_u(x) = \Delta_u(x) \cap \bar{\Delta}_u$ where $\Delta_u(x)$ is a set of possible decisions without the constraints, i.e.

$$\Delta_u(x) = \{u \in U : \bigwedge_{j \in \overline{1,k}} H_j(u_j, x_j) \leq \alpha\}.$$

Denote by $S_n = (u^{(1)}, u^{(2)},...,u^{(n)})$ the sequence of the decisions, by $\bar{u}^{(i)}$ the allocation $u^{(i)}$ for which $T^{(i)} \leq \alpha$, by $\hat{u}^{(i)}$ the allocation $u^{(i)}$ for which $T^{(i)} > \alpha$, and introduce the following sets:

$$\bar{D}_x(n) = \{x \in X : \bar{u}^{(i)} \in \Delta_u(x) \text{ for each } \bar{u}^{(i)} \text{ in } S_n\},$$

$$\hat{D}_x(n) = \{x \in X : \hat{u}^{(i)} \in U - \Delta_u(x) \text{ for each } \hat{u}^{(i)} \text{ in } S_n\}.$$

As the estimate of $\bar{x}$ one may propose the set $\Delta_x(n) \triangleq \bar{D}_x(n) \cap \hat{D}_x(n)$ and under some general assumptions one can prove the convergence of $\Delta_x(n)$ to $\{\bar{x}\}$ (a singleton). The estimate $\Delta_x(n)$ and the proof of the convergence are based on the results for the general formulation of the relational knowledge representation and the decision problem presented in [5]. In our case the knowledge representations and the results have a specific form. It is worth noting that the estimates of x_j may be determined independently for separate processors:

$$\Delta_{x,j}(n) = \bar{D}_{x,j}(n) \cap \hat{D}_{x,j}(n)$$

where

$$\bar{D}_{x,j}(n) = \{x_j : H_j(\bar{u}_j^{(i)}, x_j) \leq \alpha \text{ for each } \bar{u}_j^{(i)} \text{ in } S_n\},$$

$$\hat{D}_{x,j}(n) = \{x_j : H_j(\hat{u}_j^{(i)}, x_j) > \alpha \text{ for each } \hat{u}_j^{(i)} \text{ in } S_n\}.$$

The value $x_j^{(n)}$ may be chosen randomly from $\Delta_{x,j}(n)$. It is possible to present the estimation algorithm in a recursive form and consequently the **allocation algorithm with learning** has the following form (for $n > 1$):

1. Apply the allocation $u^{(n)}$ and measure the execution times $T_j^{(n)}$ for each processor.

2. If $T_j^{(n)} \leq \alpha$

   Knowledge validation for $u_j^{(n)} = \bar{u}_j^{(n)}$
   Prove whether

   $$\bigwedge_{x_j \in \bar{D}_{x,j}(n-1)} [H_j(u_j^{(n)}, x_j) \leq \alpha].$$

   If yes then $\bar{D}_{x,j}(n) = \bar{D}_{x,j}(n-1)$. If not, determine the new set $\bar{D}_{x,j}(n)$.

   Knowledge updating for $u_j^{(n)} = \bar{u}_j^{(n)}$

   $$\bar{D}_{x,j}(n) = \{x_j \in \bar{D}_{x,j}(n-1) : H_j(u_j^{(n)}, x_j) \leq \alpha\},$$

   $$\hat{D}_{x,j}(n) = \hat{D}_{x,j}(n-1).$$

3. If $T_j^{(n)} > \alpha$

   Knowledge validation for $u_j^{(n)} = \hat{u}_j^{(n)}$
   Prove whether

   $$\bigwedge_{x_j \in \hat{D}_{x,j}(n-1)} [H_j(u_j^{(n)}, x_j) > \alpha].$$

   If yes then $\hat{D}_{x,j}(n) = \hat{D}_{x,j}(n-1)$. If not, determine the new set $\hat{D}_{x,j}(n)$.

   Knowledge updating for $u_j^{(n)} = \hat{u}_j^{(n)}$

   $$\hat{D}_{x,j}(n) = \{x_j \in \hat{D}_{x,j}(n-1) : H_j(u_j^{(n)}, x_j) > \alpha\},$$

   $$\bar{D}_{x,j}(n) = \bar{D}_{x,j}(n-1).$$

4. Determine $\Delta_{x,j}(n) = \bar{D}_{x,j}(n) \cap \hat{D}_{x,j}(n)$.

5. Choose randomly $x_j^{(n)}$ from $\Delta_{x,j}(n)$.

The steps 2, 3, 4, 5 should be performed for $j = 1, 2,..., k$.

6. Put $x^{(n)} = (x_1^{(n)},...,x_k^{(n)})$ into the set $\Delta_u(x^{(n-1)})$ in the place of x_{n-1}.

7. Choose random the decision $u^{(n+1)}$ from the set $D_u(x^{(n)}) = \Delta_u(x^{(n)}) \cap \overline{\Delta}_u$.

The block scheme of the expert control system for the control of the computational tasks allocation with the learning process is presented in Fig. 2. The controlling part consists of the basic program generating the solution $D_u(x^{(n)})$, the knowledge base, the learning program (the validation and updating of the knowledge) and the generators of random numbers G_1 and G_2 for the random choosing of $x^{(n)}$ from $\Delta_x(n)$ and of $u^{(n)}$ from $D_u(x^{(n-1)})$, respectively.
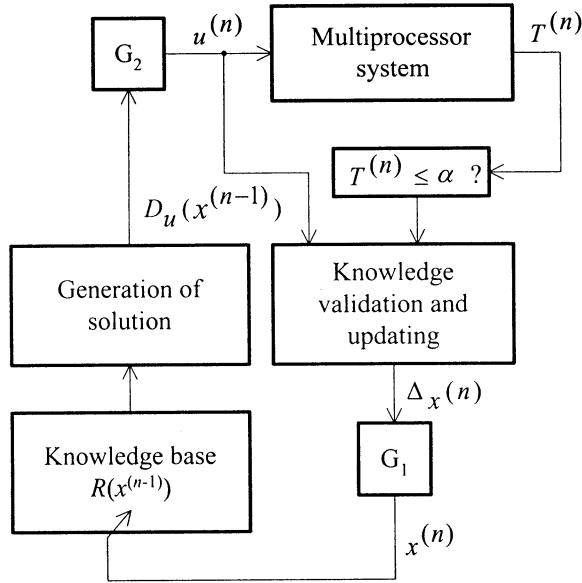
Figure 2. Expert control system with learning.

For (11) in the form (1) we obtain

$$\overline{D}_{x,j} = [0, x^{(n)}_{max,j}], \qquad \hat{D}_{x,j} = (x^{(n)}_{min,j}, \infty),$$

$$\Delta_{x,j}(n) = (x^{(n)}_{min,j}, x^{(n)}_{max,j})$$

where

$$x^{(n)}_{max,j} = \frac{\alpha}{\max_i \overline{u}^{(i)}_j}, \qquad x^{(n)}_{min,j} = \frac{\alpha}{\min_i \hat{u}^{(i)}_j}.$$

The steps 2 and 3 of the learning algorithm are now as follows:

2. If $T^{(n)}_j \leq \alpha$ (i.e. $u^{(n)}_j = \overline{u}^{(n)}_j$), prove whether

$$u^{(n)}_j \leq \frac{\alpha}{x^{(n-1)}_{max,j}}.$$

If yes then $x^{(n)}_{max,j} = x^{(n-1)}_{max,j}$. If not then

$$x^{(n)}_{max,j} = \frac{\alpha}{u^{(n)}_j}.$$

Put $x^{(n)}_{min,j} = x^{(n-1)}_{min,j}$.

3. If $T^{(n)}_j > \alpha$ (i.e. $u^{(n)}_j = \hat{u}^{(n)}_j$), prove whether

$$u^{(n)}_j > \frac{\alpha}{x^{(n-1)}_{min,j}}.$$

If yes then $x^{(n)}_{min,j} = x^{(n-1)}_{min,j}$. If not then

$$x^{(n)}_{min,j} = \frac{\alpha}{u^{(n)}_j}.$$

Put $x^{(n)}_{max,j} = x^{(n-1)}_{max,j}$.

4. Conclusions

The parameters of the certainty distributions may have a significant influence on the results. Then it may be reasonable to apply an adaptation consisting in changing these parameters in the successive cycles. To decrease the computational difficulties it may be useful to apply a decomposition of the set of processors into subsets and to design a two level decision system with the allocation for the subsets on the upper level.

It is possible to apply the description of the uncertain variables in the learning system. Then the certainty distributions given by an expert should be modified during the learning process. The considerations may be extended for more complicated structures of the computational operations, i.e. for more complex cases of systems with a distributed knowledge [6].

References

[1] Z. Bubnicki, "Uncertain variables and their application to decision making", *IEEE Trans. on SMC, Part A: Systems and Humans*, Vol. 31, no. 6, 587-596, 2001.

[2] Z. Bubnicki, "Uncertain variables and their applications for a class of uncertain systems," *International Journal of Systems Science*, Vol. 32, no. 5, 651-659, 2001.

[3] Z. Bubnicki, Uncertain Logics, Variables and Systems, Springer-Verlag; Berlin, London, 2002.

[4] Z. Bubnicki, Application of uncertain variables and logics to complex intelligent systems. Proc. of the 6th International Symposium on Artificial Life and Robotics, Tokyo, Japan, Vol.1, pp. 220-223, 2001.

[5] Z. Bubnicki, "Learning processes in a class of knowledge-based systems", *Kybernetes*, Vol. 29, no. 7/8, pp. 1016-1028, 2000.

[6] Z. Bubnicki, Application of uncertain variables to decision making in a class of distributed computer systems. Proc. of 17th IFIP World Computer Congress, vol. IIP, Montreal, pp. 261-264, 2002.

Diversity in evolutionary system

Yongguang ZHANG

Institute of Systems Science, Academia
Sinica, Beijing China, 100080
yzhang@staff.iss.ac.cn

Masanori SUGISAKA

Dept. of Electrical and Electronic
Engineering, Oita University, Japan
msugi@cc.oita-u.ac.jp

Abstract

In this paper we describe the diversity in biological system as a presentation of complexity of biological evolutionary system, and then we show some result of diversity in digital evolutionary system. Comparison with each other we can find that diversity is a fundamental complexity in evolutionary system. It arises a difficult problem for statistics principle that the event with small probability could not occur in one experiment.

Key words Evolution, Diversity, Evolvability

I. Introduction

In recent years people have pay more attention to the research of evolvability of evolutionary systems, both natural world and artificial digital living systems. Through observations and experiments in digital living system Tierra we found diversity is an important part of an evolutionary process. In fact, in natural world the process is life origin → diversity → extinguishments → recreation → diversity…. We have observed the same process in evolvable digital living system. The reason of diversity occurring is due to mutation of gene with certain probability, although it is quite small. To construct model for evolution system to figure out evolvability is necessary. We think of that the diversity is a presentation of evolvability of living systems, it is complex and stochastic, here is only some observation, but not theory, more discussion are needed.

II. Biological diversity

There is some statement about biological evolution in a book on the Biological Evolution: The Biological Evolution is certain change that occurred in an interactive process between biological spices and the environment they survive in, and the invertible change occurred in their genetic system as time goes on. Also, biology book told us that there was a spices explosion in Cambrian period; it means the amount of spices had a huge increase in a relative short period of geological time scale. This is very important period for the biological life evolution in the earth because there were only small amount of spices before it. In addition, based on the discoveries in fossil biologists concluded that there were several events of wide extinguishments of spices and recreation, which shows in Fig.1. During the recreation period the amount of spices widely increase again. If no this recreation period we could not imagine that our present earth whether or not be the appearance like we see now. Modern theory names it as Diversity in biology.

There is another type of biological diversity through artificial selection and crossbreed. For example, we have many kind of chickens those come from a same ancestry by human being's domestication. This is selected evolution; artificial selection goes instead of natural selection. The results told us that any evolution is connected with diversity. Modern molecular biology explains the reason of diversity as mutation in gene with rather small probability. So, it depends on big population and huge amount of spices.

Darwen's evolutionary tree

Diversity and Recreation
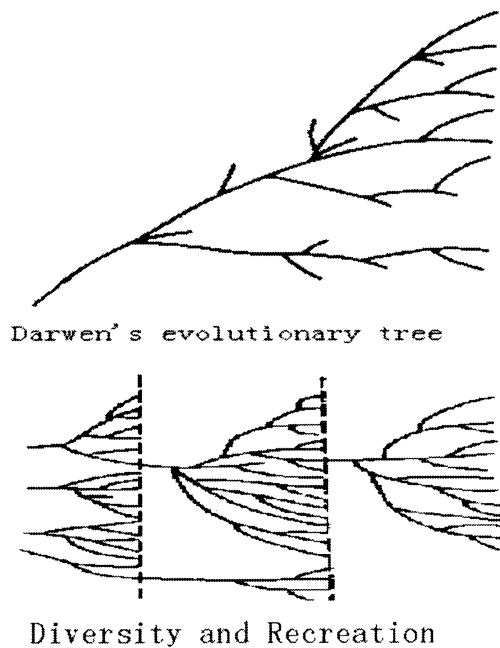
Fig.1 Darwin's evolution tree (upper) and Diversity and recreation (lower)

III. The diversity in digital evolutionary system

In the research of artificial life the evolution in digital systems is an essential problem, we concern with the evolvability. Many experiences based on Tierra have been done and we have observed some different evolutionary phenomena. In some case it was not evolvable and goes to death, in which all spices extinguished. On the contrary, some of the digital systems are evolvable, in which we have observed diversity and recreation either. An experiment result has been shown in Fig.2. The period before t < 2200 is the first part, in fact there exists very strong evolutionary action, at the beginning only one spicy lives and then it copies itself and some mutation appeared, the blue curve **nc** shows us the number of spices survive in memory at time t. At t = 0, nc=1, and then it increase very quickly till t<1000; and it keep almost the same amount of spices till t = 2200. During the period 2200 < t < 2400 seemingly there is a small extinguishments, in which the amount of spices goes down and then recreation appears, if we enlarge the local curve it would be more clear. At the period of 2400 < t < 8000, it seems keep stable, however, during the period of 8000 < t < 8400 we could found that the amount of spices goes up and then goes down quickly, it means another extinguishments period appears. What is like after t=10000? We do not know, and more experiments are needed.
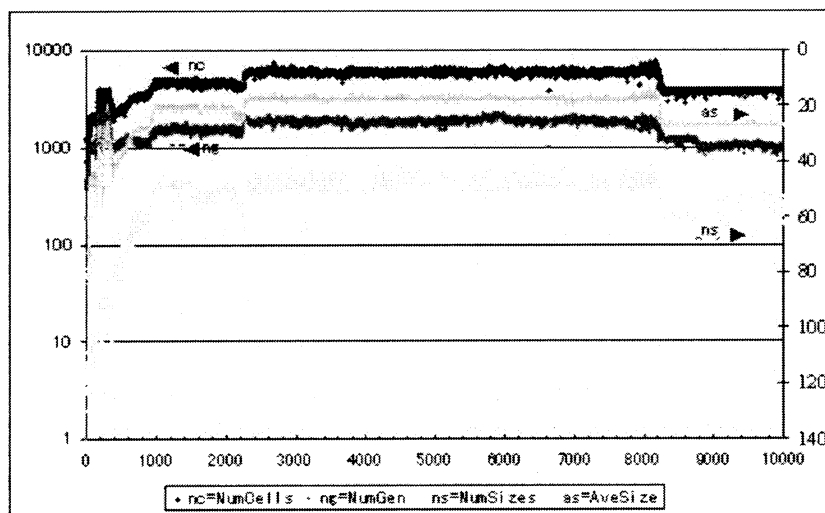
Fig.2   An experiment result on running Tierra

IV. Mutation and diversity

Basically, biologists think of the reason of diversity is from mutation in gene. Some of individuals of a spice adapted to the change in environment, and they can reproduce with gene mutated; or in an occasional some change was happened in gene of some individuals e.g. radiation, and they fortunately survived. In artificial crossbreed situation it is obviously from mutation operations. Usually, biologists suppose that the probability occurring mutation is quite small, even less than 0.001. In a long run the mutants accumulated and diversity appears, this made a preparation for next extinguishments coming. The surviving probability for any spices is not fixed. In practical it is not very big. If diversity is not enough, the remainders after extinguishments would be very small. At extremely, if no diversity then the world we live won't be the appearance like now. So, we could conclude that if no diversity, no evolution.

In the experiment shows in Fig.2 we found the same situation, the instruction of computer language is the gene of digital individuals and when the program is executing a new individual is reproduced, which maybe is the same as the original spice or become a mutant different from original one. Some of new spices can survive, but some cannot. The phenomena of diversity and extinguishments is similar to our real natural world, this means the Tierra is really an evolvable digital living system. Thus, diversity is a representation of evolvability of digital living system.

V. Question and opinion

From the statement above we believe that appearance of diversity is a very important factors to evolutionary system whatever natural or digital, and the probability of mutation is quite small. So far, our planet, the earth, is the only planet discovered in solar system and the universe in which life surviving, and it has already exists for 3.8 billions years. Although there were six or seven wide extinguishments during the long history, but there still are more than tens billion spices on it. Is it really true that the probability of mutation? Is it a very small constant? Based on point of view of probabilistic and statistical theory, some phenomena is difficult to explain. Our opinion is that the probability of mutation is not a small constant; it is variable as time change and it maybe increase obviously in some period, e.g. 0.01--0.1. Of cause, this not conclusion or theory, it is only observation and guess. More discussions are needed.

VI. Conclusion

In this paper we try to discuss the diversity appeared in evolutionary systems both natural biological world or digital living system. Diversity is an important representation of evolvability.

Based on our observations we suppose that the probability of mutation should not be a small constant, it is variable as time change and maybe increase obviously in some period. More discussions are needed.

VII. References

[1] Ray, T. S. (1994). An evolutionary approach to synthetic biology: Zen and the art of creating life. *Artificial life*, vol.1, pp.179—209. Also in: Langton, C. G. [ed.], *Artificial Life, an overview*. The MIT Press, 1995. Reprinted from: Artificial Life 1(1/2): 195-226.

[2] Ray, T. S. (1995). A proposal to create a network-wide biodiversity reserve for digital organisms. *ATR Technical Report* TR-H-133.

[3] Zhang, Y.G., Sugisaka, M., and Wu, X. Q. (1999). A bottom-up way to develop multi-cellular digital organisms. *Artificial Life and Robotics,*. Vol.4, No.3, 2000, pp.143-147

[4] Zhang, Y.G. and Shimohara, K. (2000). A note on evolvability in Tierra. Artificial Life 7 workshop proceedings on evolvability, Alife VII, Reed collega, Oregan, U.S., August1-6, 2000. pp.51--54

Control of Nonlinear Systems via State-Dependent Riccati Equation Methods

William R. Wells
University of Nevada, Las Vegas
Las Vegas, NV, USA
Email: wcube@egr.unlv.edu

Key words: extended linearization, state dependent Riccati equation, nonlinear regulation

Abstract

This paper considers an extended linearization control design technique for certain nonlinear systems. The control law is designed based on the state-dependent Riccati equation (SDRE) method. The resulting closed-loop system is asymptotically stable. These nonlinear systems can be considered as a subset of those relevant to artificial life and robotics.

Introduction

The infinite-time linear quadratic regulator (LQR) control system has received considerable attention in the literature and in application due to its ease in solution and implementation. This problem can be stated as follows [1]: Given the nth order linear system

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad (1)$$

$$x \in R^n \quad \text{and} \quad u \in R^m$$

where A and B are constant coefficient matrices, find the state x and control u which minimizes the performance index

$$J = \frac{1}{2}\int_0^\infty \left[x^T(t)Q\,x(t) + u^T(t)R\,u(t) \right] dt \qquad (2)$$

Q and R are assumed to be positive semidefinite and positive definite respectively. Also two additional conditions are imposed on the system,

1) The pair $[A, B]$ must be completely controllable

2) The pair $[A, D]$ must be completely observable where D is any $n \times n$ matrix such that

$$DD^T = Q$$

The solution for the control function $u(t)$ is given as

$$u = -R^{-1}B^T P x \qquad (3)$$

where P is the solution to the algebraic Riccati equation

$$PBR^{-1}B^T P - PA - A^T P - Q = 0 \qquad (4)$$

The Infinite-time Nonlinear Quadratic Regulator Control System

The infinite-time nonlinear quadratic regulator control system is concerned with the problem: Given the nth order nonlinear system

$$\dot{x} = f(x) + g(x)u \tag{5}$$

where $x \in R^n$, $u \in R^m$, $f(\cdot) \in C^1 and\ col\{g(\cdot)\} \in C^1$, find the state x and u which minimize the performance index

$$J = \frac{1}{2} \int_0^\infty \left[x^T Q(x)x + u^T R(x)u \right] dt \tag{6}$$

It is assumed that $col\{Q(x)\} \in C^1$, $Q(x) = C^T(x)C(x) \geq 0, R(x) > 0$ for all x. Further, it is assumed that $f(0) = 0$, $g(x) \neq 0$ for all x.

To apply the state-dependent algebraic Riccati equation method to the nonlinear system it is necessary to express equation (5) in the state-dependent coefficient form [2], [3]

$$\dot{x} = A(x)x + B(x)u \tag{7}$$

where the pair $\{A(x), B(x)\}$ is pointwise controllable in the linear sense for all x in a region Ω and the pair $\{C(x), A(x)\}$ is pointwise observable in the linear sense for all x in the region Ω.

The state dependent Riccati equation design for the feedback controller for the nonlinear regulator problem (6)-(7) is given by

$$u = -R^{-1}(x)B^T(x)P(x)x \tag{8}$$

where $P(x) \geq 0$ satisfies the algebraic state-dependent Riccati equation

$$A^T(x)P(x) + P(x)A(x) - P(x)B(x)R^{-1}(x)B^T(x)P(x) + Q(x) = 0 \tag{9}$$

Substitution of the control law (8) into (7) results in the closed-loop system

$$\dot{x} = A_c(x)x \tag{10}$$

where
$$A_c(x) = A(x) - B(x)R^{-1}(x)B^T(x)P(x) \tag{11}$$

The closed-loop matrix $A_c(x)$ is guaranteed to be Hurwitz at every $x \in \Omega$ from Riccati equation theory. Since the elements of $A(x)$ are smooth functions, expanding $A_c(x)$ about $x = 0$, and using the mean value theorem, it can be shown that the equilibrium point $x = 0$ of (10) is asymptotically stable.

Conclusions

Based upon a review of the properties of extended linearization of certain nonlinear control systems, it appears that this technique is quite relevant to the application to artificial life and robotics. Of particular interest is the control design for an infinite-time nonlinear quadratic regulator control system involving state-dependent Riccati equations. The closed-loop system resulting from this control law is asymtotically stable at the origin.

References

[1] Brogan WL (1991), Modern Control Theory. Prentice-Hall, New Jersey
[2] Friedland B (1996), Advanced Control System Design, Prentice-Hall, New Jersey
[3] Mracek CP, Cloutier JR (1998), International Journal Robust and Nonlinear Control, Vol 8, No. 4-5, pp 401-433

Representing Patterns of Autonomous Agent Dynamics in Multi-robot Systems

Jeffrey Johnson and Blaine Price
Department of Design and Innovation & Department of Computing,
The Open University, MK7 6AA, England
j.h.johnson@open.ac.uk   b.a.price@open.ac.uk

Abstract

It is proposed that vocabularies for representing complex systems with interacting agents have a natural *lattice hierarchical* structure. We investigate this for the example of simulated robot soccer, using data taken from the RoboCup simulation competition. Lattice hierarchies provide symbolic representations for reasoning about systems at appropriate levels. We note the difference between relational constructs being human-supplied versus systems that abstract their own constructs autonomously. The lattice hierarchical representation underlies both.

1. Introduction

The concept of *autonomous agents* provides an abstraction that covers both synthetic agents such as robots and biological agents, such as plants, animals, and humans. In particular it provides an abstraction that enables us to study human behaviour with a new perspective on planning and managing *socio-technical systems*.

These includes mundane systems that appear to be unpredictable, including urban and regional settlements and their transportation infrastructure, organisations and their management, and less mundane systems such as drug trafficking, organised crime, and terrorism.

There are many ways of defining agents. Ours is based on combinatorial mathematics. The basic ideas are *sets* and *relations* between sets. This includes the special case of many-one relations between sets called *mappings*, or *functions* when the sets are numbers. Agents are *represented by constructive algebraic structures*. In other words, agents are built *bottom-up*.

This is consistent with a widespread view in complexity science. What we propose that is new is how to move up the constructive algebraic hierarchy to build higher level constructs that allowing *symbolic reasoning*.

An essential feature of real systems is that the hierarchy of representation is rarely tree-like. At all level there are connections between things, requiring a *lattice-like* organisation.

Building hierarchical vocabularies takes place in the context of combinatorial explosion. Even small sets have large numbers of subsets. A set with a dozen elements has over four thousand subsets, and there are millions of ways of selecting an 11-player soccer team from a class of thirty students.

It is easy, in theory, to generate *all* combinations of anything. The research challenge addressed by this paper is how can *useful* structures be abstracted from this wealth of latent possibilities. It will be seen that this amounts to the question :

"how can we build a lattice-hierarchical vocabulary to *represent*, *plan*, *design*, and *control* the emergent dynamics of systems of interacting autonomous agents?".

In Section 2 we will develop the mathematical notation under-pinning our approach. But we do not expect mathematics by itself to give answers to this question. We believe that *computation* is another essential ingredient in complexity science.

In Section 3 we will illustrate our theory and ideas using the simple illustrative example of simulated soccer-playing robot agents. Although simple in concept, this system can generate great complexity. Understanding and controlling this system has attracted the attention of some of the world's best and most advanced researchers.

In Section 4 we discuss pattern recognition and the lattice hierarchy, and the possibility of automatic construct abstraction.

Section 5 presents our conclusions, including the lessons to be learned from the study of soccer-playing agents, and how this might generalise to other systems.

2. Mathematical Preliminaries

Let S be a *system*. S has a set of *primitives*, X = $\{\, x_1, \dots , x_n \,\}$. A *relation*, r, on X, is defined by a proposition p_r: $(x_{r0}, x_{r2} \dots x_{rm}) \to$ **T**, where **T** is a *truth set*, and $\{\, x_{r0}, x_{r2} \dots x_{rm} \,\} \subseteq X$. For simplicity in the first instance, let **T** = {True, False}. This can be extended to include probabilistic logic and fuzzy logic. We will say that the elements $x_{r1}, x_{r2} \dots x_{rm}$ are *r-related* if and only if p_r: $(x_{r1}, x_{r2} \dots x_{rm}) =$ True. When this holds we will say the system contains the object $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$, which will be called a *simplex*. By an abuse of language, we will write r:$\{\, x_{r0}, x_{r2} \dots x_{rm}\,\} \to \langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$, and say the relation r *maps* the set $\{\, x_{r0}, x_{r2} \dots x_{rm}\,\}$ to the simplex $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$.

If the primitives are said to exist at *Level N*, then the simplex $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$ will be said to exist at *Level N+1*.

Sometimes the object $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$ has a *symbolic name*. Let *Names* be the set of symbolic names. Then there is a *naming mapping*, n: $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle \to$ name. The mappings r and n can be combined in a way that maps the set directly to the name of a structured object, nr: $\{\, x_{r0}, x_{r2} \dots x_{rm}\,\} \to$ name. This notation is useful in drawing simplified diagrams, but it also allows that the structure $\langle\, x_{r0}, x_{r2} \dots x_{rm}\,;\, r\,\rangle$ can always be constructed, given the set and relation.
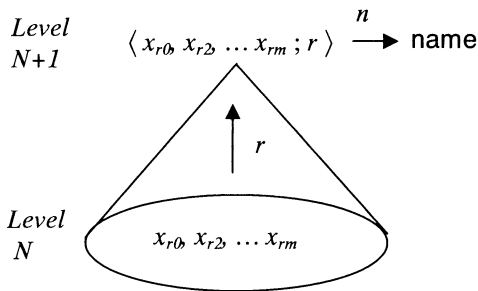
Figure 1. A set of part relationally mapped to a named simplex

The simplex, as structure, may have *emergent properties* nor possessed by its component elements. These properties may be structural. For example, an aeroplane can fly, when none of its parts can, and a choir can sing a harmony that individuals cannot. Emergent properties may also be numerical. For example, a car can have acceleration, while its parts cannot.

The generality of hierarchical structure is that parts are assembled into wholes, and wholes are assembled into higher level wholes. It is also possible for parts to be shared between higher level wholes, as illustrated in Figure 2. In this case the multi-level structure is called a *lattice hierarchy* (Figure 2).
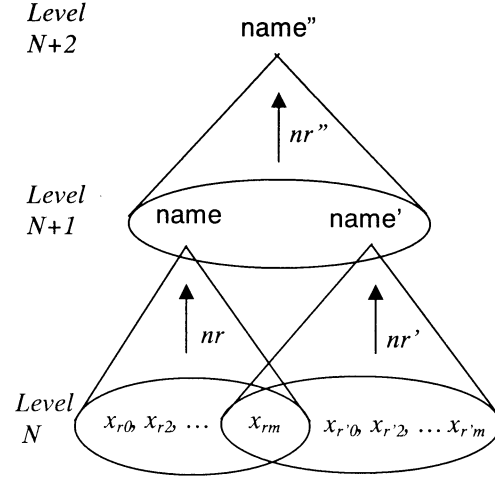
Figure 2 A multilevel lattice hierarchy

3. An example: soccer-playing robot agents

Robot soccer has taken over from computer chess as a benchmark problem in AI, Artificial Life, and Robotics. The International RoboCup Federation has encapsulated the challenge as "having a team of humanoid robots beat the human world champion soccer players by 2050". This challenge unpacks into many engineering challenges related to materials, electronics, sensing, bio-engineering, power and control. It also unpacks into the challenge of devising tactics and strategy for soccer-playing robot agents.

One of the RoboCup competitions involves a soccer simulation in which teams can control the actions of simulated players, subject to their imperfect local perception of the pitch, other players, and the ball. The great contribution made by RoboCup is that all participants must share their research findings, and in recording the many simulation games that have been played.

This is the database used for the research outlined in this paper. In the simplest case, the record for a single game is six thousand sets of twenty-three x-y co-ordinates, giving the positions of the twenty-two players and the ball. Each set corresponds to one tenth of a second, with games running for two halves of five minutes each.
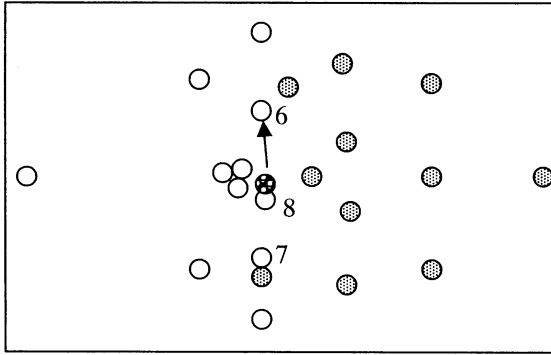
(a) Player 8 passes the ball to player 6
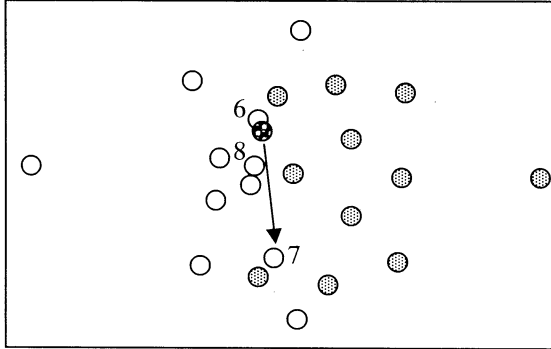
(b) Player 6 passes the ball to player 7
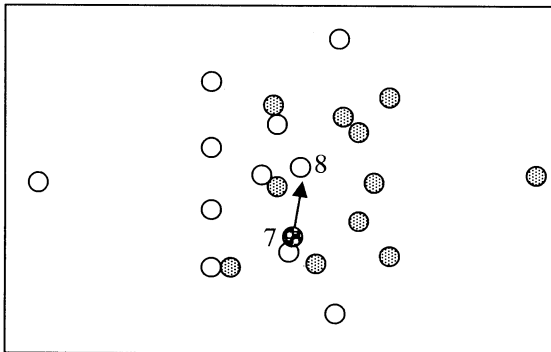
(c) Player 7 passes the ball to player 8
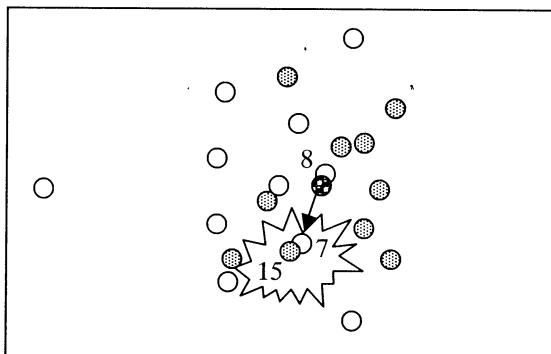
(d) Player 8 passes the ball to player 7, but it is won by the nearby player 15

Figure 3 Ball-passing relations

There is a lot of other information available, but this simple subset is sufficient for our purposes here. It is easy to write a computer program to display these data, and replay them as animations.

Following our methodology, there is a well-defined set of agents, namely twenty-two players and the ball. These can be distinguished as belonging to three subsets, the *red team* (agents 0-10), the *blue team* (agents11-21), and the ball (agent 22).

The methodology then suggests that we seek 'interesting' relationships between the agents to form higher level constructs that will be useful in reasoning about the tactics and strategy of the game.

What makes an 'interesting' relationship in this system? Presumably a pass from one member of the team to another is interesting. Figure 3 shows a sequence of passes, from robot 8 to 6, from 6 to 7, from 7 to 8, and from 8 back to 7, who then loses the ball to the opposition robot 15.

In our terms, each pass can be represented as a relationship between two robots, for example, $\langle$ robot-8, robot-6; r_{pass} $\rangle$, which can be simplified to $\langle$ 8, 6 $\rangle$. The following table can be constructed from the data file for this game.

clock tick	robot acquiring ball	pass
42	8	
54	6	$\langle$ 8, 6 $\rangle$
66	7	$\langle$ 6, 7 $\rangle$
74	8	$\langle$ 7, 8 $\rangle$
81	7	$\langle$ 8, 7 $\rangle$
82	15	

Table 1. Possession and ball passing

Apart from passes between two robots being 'interesting', sequences of passes are also potentially interesting. A *passing sequence* can have one of three outcomes: an opposing player wins the ball, the ball goes out of play and the opposition win control of the ball, or a goal is scored. Each of these can be considered to be 'interesting' *events*.

In previous AROB papers (see references) it has been argued that structural events like these can be important in devising tactics and strategies for team robot behaviour. In

particular, sequences of passing events characterise human football, as defenders are 'drawn out of position' by combinations of attacking players. Thus in terms of representation, we have individual players at Level N, pairs of player related by passing at Level N+1, and sequences of passing pairs at level N+2, as shown in Figure 4.

Pass Sequence #1
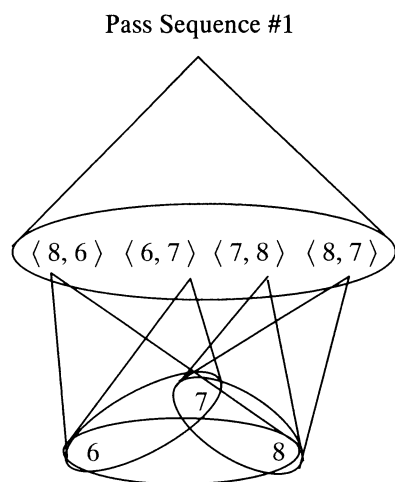
Figure 4 A lattice hierarchy of ball-passing

The game quoted in Figure 3 and Table 1 above was the 2000 RoboCup Simulation Final, won 1-0 by Portugal (the 'red team'). A list of the numbers of passing events for the first half of this game is given below.

Run Sequence # passes	Red Team	Blue Team
7	1	0
5	1	1
4	2	0
3	3	3
2	5	5
1	11	8

Table 2. Ball-Passing Sequence Events

From the table it can be seen that the red team dominates the game in terms of having the most and the longest passing sequences. Certainly when one watches this game, the red team seems to dominate. As in human football, the ability to pass accurately is very important in robot soccer. It is also important that the ball is passed to a player who is in a 'good position' to do something with it.

The leads to another canonical relationship between the soccer robots, the 'closest to'

relation. At any tick of the clock it is possible to compute the distances between the robots, and for each find that which on the other side is closest.
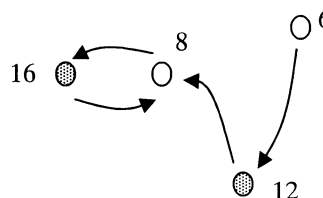
Figure 5 The closest opposing robot relation

This relationship is generally not symmetric. For example above, robots 8 and 16 are mutually closest to each other, but robot 12 is closest to robot 8, while robot 6 is closest to robot 12.

In human soccer, defenders are sometimes given the task of *marking* another player. This means that they have to stay close to that player. This usually results in a symmetric 'closest to' relationship between the two player, that can be very frustrating for the attacking player. Unless the defender can be 'shaken off', by interaction with other team members. In other words, a player forced in to the structure ⟨ me, marking opponent ⟩ might seek to form a structure with ⟨ ball, team-mate⟩ in order to break the me-marker relationships, as illustrated in Figure 6.

(a) A is closely marked by B

(b) B is drawn to C and the ball

(c) A has shaken off B
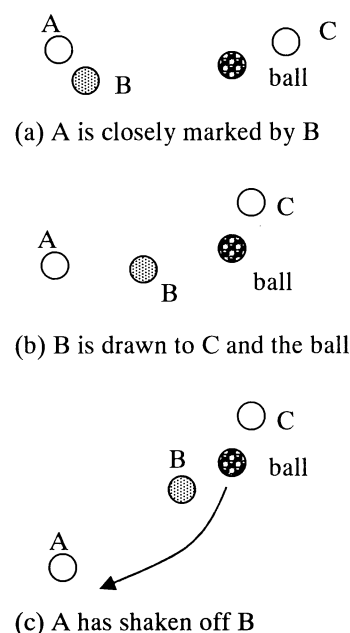
Figure 6 Shaking off a close marking opponent

4. Pattern Recognition and the Lattice Hierarchy

The previous section illustrated the formation of structures representing constructs such as 'passing' or 'being closest. To find examples of these in a data set it is necessary to *recognise* the defining *pattern*. This means:

(i) appropriate sets of candidate elements have to be recognised for the relation r
(ii) the relationship r has to be tested between the elements of the candidate subsets

For example, in recognising 'closest to' structures, it is necessary to generate all the pairs (A,B) where A belongs to one team and B belongs to the other. Then it is necessary to test to see if A is closest to B. Although this appears to be a binary relation, it is more than this. To decide the closest opponent to A requires that *all* the opponent pairs are formed and the emergent property of distance(A,B) be calculated. The particular $\langle$ A, B; $r_{closest}$ $\rangle$ is therefore recognised by inspecting the whole set of pairs, (A,B). Thus $\langle$ A, B; $r_{closest}$ $\rangle$ is higher in the lattice hierarchy than the pair $\langle$A, B$\rangle$; where A is a red robot and B is blue.

In terms of implementation, (i) usually involves forming lists, where each element can be tested independently of the others. It just has to be tested to see if it has the required properties. On the other hand, (ii) can be much more demanding, since it may require that relationships between all the elements be tested simultaneously. This may involve complicated functions to build constructs and test them.

One of the great objectives in building intelligent agents is to have agents that can structure the universe for themselves, by abstracting their own constructs. Although there are combinatorially many subsets, generating testing any given subset is not always a computationally demanding task, especially when notions of sampling are used. By contrast, the possibility of generating and testing 'useful' relationships is much more onerous.

In the first instance we are analysing the RoboCup data using human-inspired constructs such as the pass-sequences and closest-opponent as discussed above. However, the data lend themselves to more open-ended experiments in automatic construct generation. In our terms, means generating the lattice hierarchy and generating a *vocabulary*

to name it structures the various emergent levels. It also means generating automatic pattern recognisers to test elements and relationships.

Conclusions

The lattice hierarchy is, arguably, one of the simplest structures available for representing complex systems with hierarchical vocabularies. To use this symbolic representation it is necessary to have pattern recognisers between the levels, able to determine higher level structures. Hierarchical aggregation through relatively simple substructures is likely to be computationally the most tractable. Ultimately, synthetic systems must abstract their own constructs and vocabularies. We believe that in all cases, the lattice hierarchy will implicitly or explicitly be used.

These ideas generalise to other complex system, including the social systems mentioned in Section 1. Although robot soccer, even in its simulated form, is difficult to control successfully, it is much simpler and better behaved than many human systems. For this reason we think it is an appropriate laboratory subject for understanding better the nature and use of lattice hierarchies.

References

RoboCup Federation, (www.robocup.org) Proc. RoboCup Conferences, 1998, 1999, 2000, 2001, 2002.

Johnson, J.H., Sugisaka, M., 'Desirable-Event subgoal theory in the design of swarm robot control systems, AROB-7, Japan, Jan 2002

Johnson, 'Visual communication in swarms of intelligent robot agents', Artificial *Life and Robotics*. 2001

Johnson, J.H., Sugisaka, M., 'Complexity Science for the design of swarm robot control systems', *Proc. IECON-2000*, IEEE Conf. on Industrial Electronics & Control, 2000

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

Johnson, J.H., 'Robot football, artificial life, and complexity', *Artificial Life and Robotics*, **3**, 230-235, 1999

Generalized Artificial Life Race and Model

Tu Xu Yan

Hon. President, Advisory Committee Chairman of Chinese Association of Artificial Intelligence,
Professor, Advisor of School of Information Engineering, University of Science and Technology Beijing.
Tel: 86-10-62333554
Email: tuxuyan@vip.sina.com

Abstract:
In this paper, the concept, race and model of Generalized Artificial Life (GAL) are given and discussed.

Keywords:
Artificial Life(AL), Generalized Artificial Life(GAL), KVP Model

1. Introduction

In October of 2002, The National Symposium on "Artificial Life(AL) and its application" was organized by Chinese Association for Artificial Intelligence(CAAI) and held in University of Science and Technology Beijing. The main subject is "Generalized Artificial Life (GAL) and its Application".

The GAL is the generalization and development of AL:

* The GAL system is the man-made system with the similar vitality of Natural Life(NL) system. The vitality means not only external phenomena and behavior, but also internal attribute and function.

* The GAL System could be made by engineering technology way, such as computer software, electronic, mechanical and optical hardware, or by biological science way, such as clone technology, gen-transfer method.

* The GAL is not only the simulation of NL, but also the extension or expansion of NL, especially the extension of Human Life(HL). For example, the artificial organ could be instead of the ill human organ.

* The research and development of GAL require both the "synthesis" and "analysis" methodology.

In our opinion:

- *Life come from biology.*
- *Life exist in system.*
- *Life based on information*
- *Life essence is intelligence.*

So that:

Artificial Life is the product of the combination of biology, system, information and intelligence science and technology.

In this paper, the concept, race and model of GAL are given and discussed.

2. Race of GAL

The race of GAL system is shown as Fig 1.
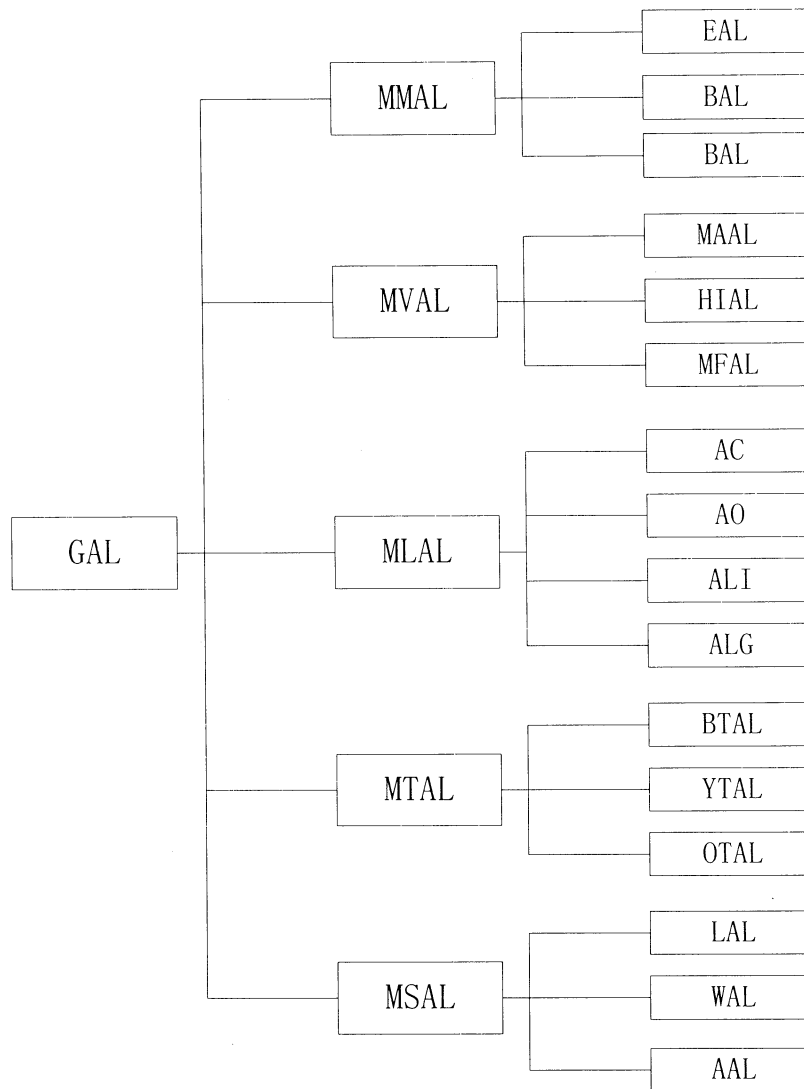
Fig 1. Race of GAL

In Fig 1:

1. MMAL (Multi-media AL)

The MMAL could be made by·multi-media and multi-methods, such as:

• EAL (Engineering AL)

    For example: Digital Life, Intelligent Robot.

• BAL(Biological AL)

    For example: Clone animal, Gene-transfer animal.

• BEAL(Bio-Engineering AL)

    For example: Cyborg, Bio-Electronic robot.

2. MVAL(Multi-vitalities AL)

The MVAL is the AL with multi-vitalities simulated NL. Such as:

•   MAAL(Multi-Attribute AL)

For example: autonomy, activity, sensitivity, reactability, flexibility, mobility sociality, etc.

- HIAL (High-Intelligence AL)

For example: Self-adaptability, Self-learning, Self-organization, Self-evolution, Self-growth, Self-reproduction, Self-stabilization, Self-optimization, Self-coordination, etc.

- MFAL (Multi-Function AL)

For example: Perception, thinking, motion, stepping, swimming, flying, eating, mating, working, etc.

3. MLAL (Multi-levels AL)

The MLAL include various levels of AL, such as:

- AC (Artificial Cell)

For example: E-Cell, clone cell.

- AO(Artificial Organ)

For example: artificial heart, kidney, lung; artificial eye, ear, nose; artificial limb, joint etc.

- ALI(Artificial Life Individuality)

For example: humanoid robot, artificial fish, clone sheep, mechanical cat, gen-transfer mouse, etc.

- ALG (Artificial Life Group)

For example, artificial society model, artificial ecology system, artificial fish society, intelligent robot team, etc.

4. MTAL (Multi-Time period AL)

- The MTAL includes different time period of AL, such ar:
- BTAL (Baby Time period   AL)
- YTAL (Youth Time period AL)
- OTAL (Old Time period AL)

5. MSAL (Multi-Space range AL)

The MSAL include various AL in different space range, such as:

- LAL(land AL)

For example: artificial sheep.

- WAL(Water AL)

For example: artificial fish.

- AAL (Air AL)

For example: artificial bird.

3. Model of GAL

In order to research and develop GAL system, The model for analysis and synthesis of GAL system is required.

By means of the GMM(Generalized Modeling Methodology) in LSC(Large System Cybernetics). Based on the integration of knowledge model and mathematical model, The KVP model for GAL System is given as follow:

GALs={K[n, m, w, l],V[a, f, b, p],P[x, x, y, u, C(t, s)]}

(1)

Where:

GALs-Life System

K—Kinds of Life System

V—Vitality of Life System

P—Process of Life System

The KVP model is a multi-layers, multi-elements, dynamical generalized model. It is integrated by the following submodels:

1. Kinds Model of Life

The Kinds model of life is a tuple as follows:

K=K[n, m, w, l]   (2)

Where:

n—name of life system

Such as: human, animal, plants, microbe etc.

m—media of life system

For example: protein, non-protein.

w—way for life system

Such as: computer software, clone technology, etc.

l—level of life system

For example: cell, organ, individuality, group etc.

2. Vitality Model of Life

The Vitality model of life can be shown as a tuple:

V=V[a, f, b, p]   (3)

,where:

a—attribute of life

Such as: autonomy, activity, sensitivity, reactivity, flexibility, mobility, sociality etc.

f—function of life

For example: self-adaptability, self-learning, self-organization, self-evolution, self-growth, self-reproduction, self-stabilization, self-optimization, self-coordination, etc.

b—behavior of life

Such as: thinking, perception, motion; stepping, swimming, flying; eating, mating, working, etc.

p—phenomena of life

For example: bearing, olden, illness, death.

3. Process Model of Life

The life system is a living non-linear dynamic system with varying structure and parameter. The life is the dynamical process of life system from bearing to death.

The process model of life can be presented by a non-linear varying coefficient differential equation as follows:

$$p[x, \dot{x}, y, u, c(t, s)] = 0$$ (4)

where:

x(t, s) —state of life system

$\dot{x}(t, s)$ —variation of state, $(\dfrac{\partial x}{\partial t}, \dfrac{\partial x}{\partial s})$

y(t, s) —output of life system

u(t, s) —input of life system

c(t, s) —varying coefficients

t—time

s—space

The KVP model for life system is an integrated frame of models. It can be used for integration of the existed models, such as, cellular automata, L-system, genetic algorithm, evolutionary programming, chaos, fractal etc. , and applied to establish the now model for life system.

The KVP model is proposed for GAL system. However, It can be applied to represent both artificial life and natural life system.

4. Conclusion

In this paper, the race and model of GAL are given. We believe:

The race of GAL is helpful to open up the domain of research and application of AL for simulation, extension and expansion of NL.

The KVP model frame can be used to integrate the existed models and to establish the new model for life systems both artificial life and natural life systems.

Main reference

1. Tu Xu Yan and Yin Yi Xing .ed, <Artificial Life and Applications>, proceedings of CAAI-Symposium on Artificial Life and Applications. Oct. 2002. Beijing university of Science and Technology Beijing.

2. Tu Xu Yan, "Generalized Artificial Life and its Applications", <Artificial Life and Applications>, Oct. 2002. Beijing

3. Tu Xu Yan, "Generalized Artificial Intelligence" < Progress of Artificial Intelligence in China>, Proceedings of 9th CAAI Conference, Beijing University of Posts and Telecommunications Publishing House, 2001, Beijing.

4. Tu Xu Yan. "AI, AL and Robotics", Proceedings of FIRA World Congress, (Plenary Speech), Seoul, Korea, 2002.

5. Tu Xu Yan, "Generalized Intelligence System", Proceedings of Korea-China Joint Workshop on Intelligent Systems, Seoul, Korea, 2002.

6. Tu Xu Yan, <Large System Cybernetics>, Defense Industrial Publishing House, 1994, Beijing.

7. Tu Xu Yan, <Artificial Intelligence and its Applications>, Electronics Industrial Publishing House, 1988, Beijing.

8. Tu Xu Yan <Bio-Cybernetics>, Science Publishing House, 1980, Beijing.

Development of a self-driven personal robot

Yoshihiro Takenaga and Eiji Hayashi
Faculty of computer science and systems engineering, Kyusyu Institute of Technology
680-4, Kawatu, Iizuka-City, Fukuoka Prefecture, Japan

Abstract

In recent years, production technology has become a typical application area for robots, and mechatronics has decreased various types of human workloads. Moreover, robotics technology has made remarkable progress in helping to shoulder the human burden of precise work. With regard to communication between humans and machines, users can accurately control and operate robots used in production technology by means of numerous operation commands and keys. However, at present, these operations are somewhat complicated for the general user. We have thus attempted to develop a personal robot that can move autonomously.

1. Introduction

In recent years, the rapid development of semiconductor technology has resulted in dramatic advances in computer processing speed and memory capacity. These advances have made it possible to control and process robots to have high degrees of freedom and multiple information inputs from sensors and CCD cameras.

Sony Corporation's AIBO robot is already on the market, as is Honda Corporation's ASIMO. The creation of a robot that uses such techniques constitutes a high level of progress achieved by means of development and marketing. The strong consumer demand for the AIBO has heightened concerns regarding the coexistence of robots in society, thus necessitating this issue's further study. In Japanese society in the future, the proportion of nuclear families in relation to extended families will increase, and at the same time the elderly will account for the largest share of the population. Together, these trends mean that more elderly people will be living alone, without the aid of their children or grandchildren. So even while it becomes more important to help the elderly in their home lives, there will also be a shortage of people available to provide such help. Therefore, robots are needed to provide such help.

The aim of this study is to develop a personal, self-driven robot. The goals of this robot are to perform simple tasks that are very easily performed by humans; for example, bringing a cup or a newspaper, or turning a television or other device on and off. In this paper we describe the mechanism, control, and information processing of such a robot.

2. Composition of personal robot

The robot that is the subject of this study has already been used in factories. This type of robot is physically efficient at high output, and works at high speed and with great precision. However, the configuration of this type of robot is too difficult for humans to operate without advanced skills and knowledge.

Communication between the human operator and the robot is important. Because the personal robot "lives" in the house with humans, situations in the living area and the tasks that the humans require will change.

In the future, the personal robot will have to operate autonomously and be required to meet the demands humans place upon it, not only in the home but in various environments including offices and sickrooms.
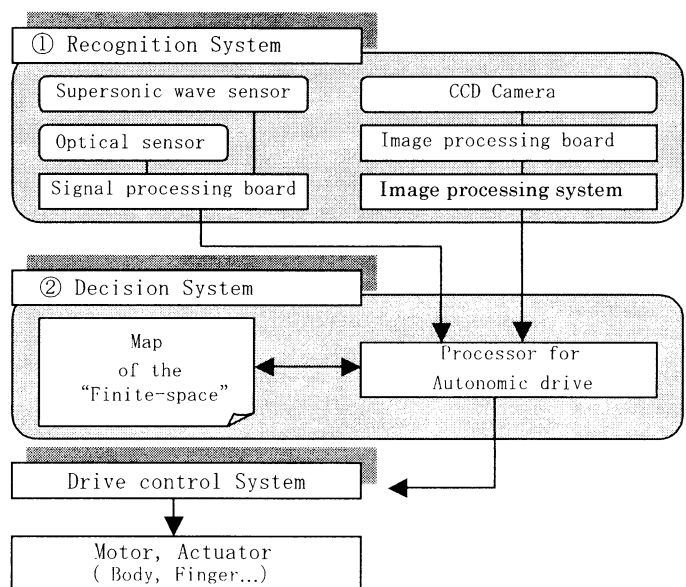
Fig. 1 Processing system for the personal robot.

Robots of the present are unable to meet these kinds of demands. We have suggested that in order for a robot to operate autonomously, it should be set up in a one-room

environment such as those described above (referred to as "finite spaces"). In such a space, the robot is capable of moving autonomously, though at the user's direction, to a desired location. This study undertakes the development of a system for basic functions, such as spatial recognition allowing it to understand its environment, movement processing based on the finite space map, and revision of a path using a visual system. The system and the processing composition are shown in Figure 1. The spatial recognition faculty of the ① is composed of a visual system that lets the robot see or understand its environment, as well as a sensor combination system. The operation determination faculties of the ② are the systems that determine the robot's action, direction, and speed via feedback input from the space recognition faculty and the feedforward input to refer to the finite spatial map. The design of the robot is shown in Figure 2.
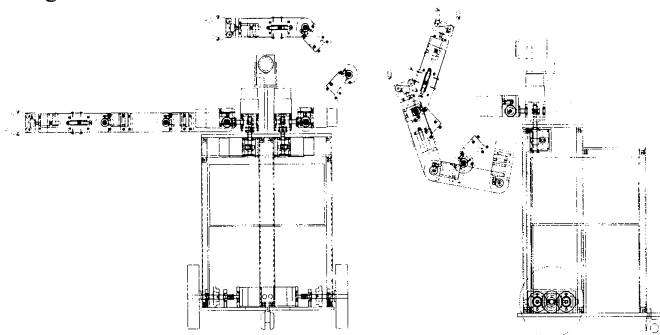
Fig. 2　Robot design

3. Robot Mechanisms

3.1 Drive mechanism

The size of each of the robot's parts was determined based on the size of an ordinary Japanese-style house. The frame and chassis include space to hold a battery, a computer, and an amplifier.
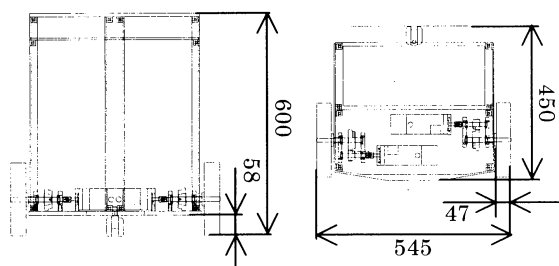
Fig. 3　Design of the drive machinery

The drive mechanism has two independent driving wheels and a caster. The front wheel has two independent driving wheels, and the rear wheel has one caster. Therefore the robot has a small turning circle. And going up and down

of a difference in level was taken no thought. Shifts between horizontal levels is not a matter of concern since the limited space in which the robot is to be used is assumed to be barrier-free. The drive mechanism of the robot is shown in Figure 3.

3.2 Arm

The arm has six joints because the work area must be wide and must accommodate complicated work. The link composition is comprised of a three-link system. The shoulder has three degrees of freedom. Each joint has one degree of freedom. At the wrist, a motor turns the hand. A stepping motor is fixed to the inside of each link in company with gear wheels. Each joint is driven by a worm gear and a worm wheel. Because the mechanism is simple, its space efficiency is high, and its position control is very precise. However, the arm's ability to practically manipulate weight decreases with the larger weight of the stepping motor. This, however, does not matter because the lifting work to which the robots is to be applied is relatively light.

An object to be handled weighs about 500 grams. Work was limited simple tasks; for example, getting a newspaper and cup, turning a switch on and off and so forth.

When the arm is idle, it is placed at the rear of the body in order to keep the robot compact.
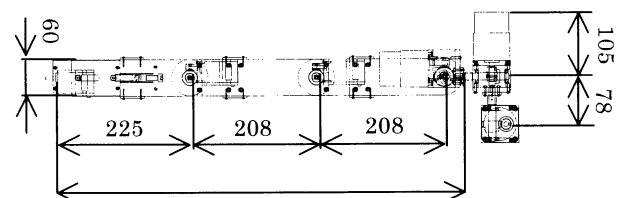
Fig. 4　Arm design (side view)

4. Experiments and efficiency evaluation

4.1 Efficiency evaluation of the drive mechanism

The motor was given the following commands:
① Go straight

② OF drive on one

A gap amount

Drive

Sarvobrake
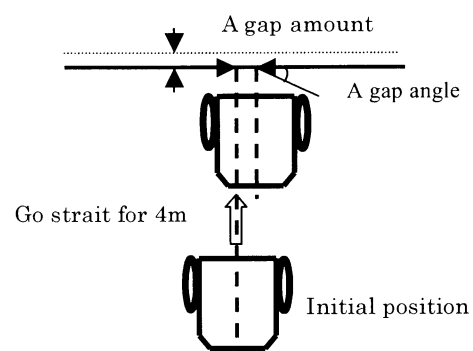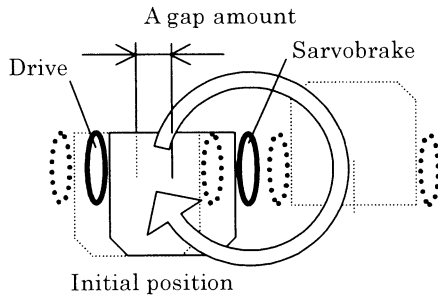
Initial position

③ A circuit command to drive two wheels in opposite directions

A gap amount

Later position

Initial position

Drive
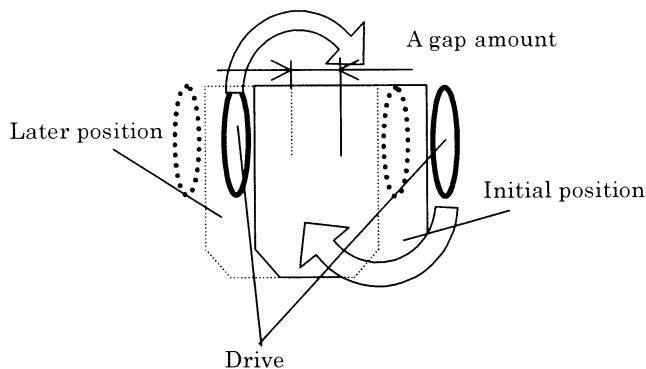
In the case of ①, the gaps were measured after going straight for 4m. In the cases of ② and ③, the gaps were measured from the initial position after the circuit was completed.

The distances of these gaps are given to the robot as fundamental data. The robot refers to these data when it drives. It also refers to information acquired by the CCD camera.

4.2 Efficiency evaluation of the arm

The maximum bend angle and the revolution angle of each arm joint were measured. These data become the range within which each joint can move. When the arm was rotated 180 degrees on a shoulder, the amount by which each link slipped was measured. The robot considers this the gap size and controls the arm using these data.

4.3 Orbit route

For the sake of efficient data processing, the robot is given a fundamental orbit route for its arm.

For example, when the robot goes to get an object on a desk, the orbit routes to change the bend state of the arm are established as well. When it drives, the robot obeys the fundamental orbit routes.

The means by which the pulse data are given to the motor are as follows. First, the movements are separated into a few steps. Second, the amount and direction of each movement are calculated from the present bend angle to the next bend angle. Third, the joint that had to exert its maximum bend angle is given the maximum pulse. The other joints is given pulses in proportion to the ratio between the angle needed and the maximum angle. The pulse data are composed of numbers of steps. The pulse data are composed of several orbit routes, and the robot was driven in the experiment.

5 Conclusions

A robot structure intended to achieve a necessary level of efficiency was developed in this research effort. As a result, we were able to satisfy the body size requirement and identify the installation position.

·   The fundamental data to be supplied to the robot were acquired, as shown by a sufficient result obtained in the experiment concerning drive structure and arm structure.
·   Efficient data processing could be performed in order to propose several orbit routes to the arm in the arm drive experiment.

Realistic data regarding handling, making orbiting, and driving around a room will be acquired in the future. The robot's hand must be developed and the structure of the head must be improved.

References

[1]Funakubo, Process and recognition of the vision paturn, Keigakusyuppan, 1990.
[2]Shinoda, Self-driven robot on the grand, Journal of the Robotics Society of Japan, Vol. 18, No.7, pp.928-932. 2000

Behavior-Based Autonomous Robotic Systems and the Reliability of Robot's Decision
The Challenge of Action Selection Mechanisms

Maki K. Habib

School of Engineering and Science, Monash University Malaysia
Bandar Sunway, Petaling Jaya, Selangor, Malaysia
maki@ieee.org

Abstract

Behavior-based systems show promising solutions that are flexible, robust, adaptable to meet real time requirements, and can deal with unexpected events in a dynamic environment. It has key properties expressed in the ability to learn, react fast, express behaviors with various representations, and with its flexible structure it can accommodate reactiveness and representation. Important issues are still facing behavior-based systems and need to have proper solutions, such as, behavior generation, how to design simple behaviors that guarantee robust operation in spite of the limited knowledge available at design time, behaviors and implicit representation, handling uncertainty, behaviors and experience, distributed approach in behavior control, multi-valued actions and action selection, and how to coordinate the activity of several and possibly competing behaviors in order to perform a complex task, etc. Behaviors typically produce a single-valued output that is viewed as best to perform an action and accordingly, there is a need to decide what action is the most appropriate for each situation. But, at some input states of the environment, several behaviors may become active and there may be multiple actions competing for execution. Also, it is possible that the generated actions by different behaviors may be contradictory and hence a mean to resolve conflict situations is needed. Accordingly, there is a need to resolve, decide and/or select/conclude the most appropriate action from a set of competing actions. A related problem to the action selection is the reliability of the decisions the robot makes.

Keywords:

Mobile Robots, Control Architecture, Behavior Based, Autonomy, Action Selection.

1. Introduction

Research on Autonomous intelligent systems normally includes disciplines spanning over almost every field in engineering and science. Intelligence should be best illustrated by systems and devices that can directly sense and act in their own environment without demanding detailed supervision from human. Intelligent autonomous robots with the ability to move are one of the most exciting and important entities that can work in real physical world. Intelligent robots are essential due to their immediate applicability in a variety of tasks to replace human being in hazardous environments such as space missions, chemical compounds, underwater, scientific exploration, and disaster areas. It can be used for inspection, office automation, household support, entertainment, and in general to work with human being to facilitate comfort and help. Mobile robot navigation is a difficult problem to solve, and it becomes increasingly difficult when a robot encounters a failure in either its sensors, devices that tell a robot about its environment, or its actuators, devices that allow a robot to physically interact with its environment. In addition, robots working in real environment offer valuable information about the uncertainties present both within the robot and the unpredictable changes in its environment.

An autonomous mobile robot must be capable to interact, navigate and adapt its behavior intelligently to meet task requirements and the requirements of any new situation that may face. Also, it must be capable to deal with uncertain, unpredictable and dynamic environments. In addition, for a robot to behave intelligently it is required to be persistence. Persistence is tightly coupled to mobility. When a robot moves from one place to another, it preserves its internal state and data. In particular, it retains any computing results obtained during execution in the execution environment it is transported from. These capabilities represent the core functionality any autonomous mobile robot should be endowed with. Real environments contain elements of variability that limit the use of prior information. An autonomous robot is required to act within a real environment that can neither be completely nor correctly modeled, and must have the ability to alter its behavior as the context within which it is acting changes. Such context is determined by the internal state of the robot, and the state of the perceived external environment. To cope with these difficulties, robot's controller must be able to respond to unforeseen events as soon as they are perceived [1-7]. Current robots have far fewer sensors and a far less sophisticated computer than human beings; consequently, they have great difficulty adapting to changes in their capabilities and environments. The problems of robot navigation are intensified if certain sensor capabilities are lost due to malfunction or damage incurred during operation. Also, imperfect sensors and the inherent difficulty of the perceptual interpretation

process, imperfect actuators, limited resources, incomplete knowledge about the environment and unpredictable environment are contributing to degrade robot's performance by negatively impacting the direct link of sensing to proper actions. Researchers have developed several approaches to handling sensor failure including the use of Bayesian networks, case-based reasoning, evolutionary learning, etc.

Autonomy requires systems that are not only capable of controlling robot's motion in response to sensor inputs by sensing the situation, but that perform actions based on their own situation's assessment of circumstances. Moreover, it should be able to overcome errors in perceptions and actions. Autonomous robots have to adapt situations and manage resources that enable them to sustain themselves over extended period of time. In some environments basic autonomy is not just an option, but it is a requirement, such as, in many underwater or planetary exploration applications. In most other environments there is a need to understand the possible or potential degrees of autonomy before having the robot ready to interact with its physical world or even with other robots. Real-time and synchrony are the further key topics in understanding dynamical systems in physical environments, while reliability and robustness are the essential constraints on the road towards practical systems [8].

In order to navigate successfully, an autonomous and intelligent robot should be able to make high-level navigation decisions as well as low-level navigation decisions. To make rational decisions through proper selection of actions a robot needs to have access to the state of the world and to its own experiences. The critical question is how to make such experience available to the robot? The presented work in this paper attempts to highlight such requirements and discusses the approaches that have been developed. The architecture needed for intelligence should reflect more direct coupling of perception to action, distributedness and decentralization, dynamic interaction with the environment and intrinsic mechanisms to cope with resource limitations and incomplete knowledge [13]. Behavior-based systems show promising solutions that are flexible, robust, adapting its behavior and can deal with unexpected events in a dynamic environment. Important issues are still facing behavior-based systems and need to have proper solutions. Among these issues are multi-valued actions and action selection, and how to coordinate the activity of several and possibly competing behaviors in order to perform a complex task. This paper will discuss the behavior base control approach and focus on the action selection mechanisms and their associated approaches while highlighting the important issues that should be taken into consideration during the design of behavior based systems and action selection mechanism.

2. Robot Control Architectures

Autonomous mobile robots have to be adaptive to different kinds of environments and have to deal with complex environments and complex task. Hence, control architecture becomes very important. Different properties of real environment and robot's task impact robot's controller and therefore the choice of control architecture. Task requirements can constrain the architecture choice.

Understanding issues related to robot's behavior and their interrelation such as, representation, specification, behavior content and time scale for action, interaction, and coordination are essential to have reference criteria for selecting a control architecture and allowing to compare and evaluate different architectures relative to specific robotic designs, tasks, etc. Such criteria may include ability to execute parallel behaviors simultaneously, runtime flexibility for adaptation and learning, time scale in relation to real time reasoning and actions, action selection mechanism, modularity and the way to use abstraction, robustness and fault tolerance, programming environment and development tools, etc.

Different architectures have been proposed and investigated [15-23]. These architectures range from symbolic artificial intelligence planners to the increasingly popular behavior-based approaches. These include deliberative planning with the sequential capabilities of sense, plan, and act with longer time scale. Reactive that maps perception of robot's word into actions systems and respond to real-time requirements of the environment. Hybrid control architecture combines layered organization with behavior based decomposition that enable to plan slowly and react quickly, i.e., combining two time-scales in an effective way. Finally, behavior-based systems attempt to bring the different time-scales closer together by distributing slower computation over concurrent behavior modules.

3. Behavior Based Control Systems

Intelligent systems should exhibit emergence property that is not designed into any of its individual sub-components. This property is important to the behavior-based paradigm to ensure the kind of robustness that traditional based systems fail to exhibit when facing the richness in the real world that cannot be captured by a priori categorization. It gives the robot the ability to adapt to different environments without explicitly programming every state-action relation.

The behavior-based approach is a methodology for designing autonomous robots and it is a way of decomposing the control strategy needed for accomplishing a robot task into units of behaviors that may includes states/memory. This approach is biologically inspired, bottom-up philosophy, and allowing for a certain freedom of interpretation. It is gaining increasing potential as new control paradigm that enables robots adapting to the dynamics of real world environments, have very good robustness and real-time properties to supports autonomous navigation. This approach addresses the fundamental Artificial Intelligent (AI) issues of sensing, thinking, and acting in real-time

and presents a successful approach to solving situated AI problems. One of the main challenges of autonomous robotics is to devise a theoretical framework for analysis, design and synthesis of systems that have specified properties, performance and behavior. In the behavior based system synthesis, a complex control task is divided into a set of simpler control problems that are implemented by behaviors.

Behaviors are connecting sensory information to actuation. It sets robot's skills for responding to the situations encountered in the environment. Behaviors are processes or control laws that are generally executed concurrently and can achieve and/or maintain the goals of the system individually or as group, thus achieving the task. Behaviors are typically higher-level than actions. In building networks of behaviors, behaviors have the ability to take inputs from sensors or another behaviors in the system and sending outputs to actuators or another behaviors in the system. Based on the activation conditions associated with the world and task, the inputs determine the activation level of a behavior. These behaviors all run in parallel and their resulting commands are combined according to a decided mechanism to give the resulting control actions. Combining primitive behaviors into one system, so that complex behavior emerges. Emergent functionality may carry with it compound problems that could parallel the difficulties in developing the system. In a behavior-based system, the goal need not be explicitly represented. Instead, behaviors are selected on an immediate sensory basis in such a way that they are likely to move the robot closer to the goal in the real world.

Behavior based systems are not limited in the ways that reactive systems are in their expressive and learning capabilities. Also, there is a key difference between behavior-based and hybrid systems. This difference can be expressed in the way representation and time-scale is handled. Keeping in mind that the distribution of representation is still limited. Behavior-based systems attempt to make the representation parallel, distributed, and active, in order to accommodate the real-time demands of other parts of the system. As a result, a behavior-based approach has key properties expressed in the ability to react in real-time, to express behaviors with various representations, and with its flexible structure it can accommodate reactiveness and representation without the need for intermediate layer.

Behavior-based systems have been effective in a variety of applications, but due to their limited use of representation or lack of abstract representation, they are not ready to be employed at a higher level and hence they have been not used for complex problems involving sequence of behaviors, such as, tasks involving temporal sequences, or hierarchical task representations. However, to date they have been used for relatively simple tasks, which do not require a complex sequencing of the robot's behaviors [9]. Also, behaviors are typically redesigned to encode the specifics of any particular task. The key challenge is in how representation in any form of world model can be effectively distributed over the behavior structure. The representation must be able to act on a time-scale that is close if not the same as the real-time parts of the system. Similarly, the representation needs to use the same underlying behavior structure as the rest of the system. Not every part of a behavior-based system needs to be involved with representational computation, i.e. behaviors can be separated into representation based and primitive ones without representation. Most of currently available behavior based system cannot have the capability to generate behaviors automatically and they have been typically constructed by hand for each task. Also, they lack of the generality that requires system redesign from one task to another even if the underlying behavior remains unchanged [10].

Current behavior-based systems in general (in spite of some efforts) do not have the capacity to predict possible future world states. So, any motivated decisions made by a behavior-based system will be made on the basis of only present and past states. An important consequence of this is that a behavior-based motivated robot may only synchronize the selection of behaviors with its motivations, not their satisfaction or the satisfaction of any longer-term goals. A behavior-based robot does not have the capacity to be as effective in satisfying its motivations as a symbol-based system. Behavior-based systems can be made more adaptive by dynamically changing the configuration of behavior set by factors other than sensed conditions in the environment, thus making a move toward increasingly cognitive means for generating complex behavior. Therefore, the ability to create and act to satisfy multiple goals, and a selection mechanism to select between these goals are important requirements.

Many of the available approaches of behavior-based approach have centralized control. Some may argue that the problem of action selection within behavior-based control can be better solved with methods that centralize knowledge about resolving conflicting actions. Distributed approaches tend to be more robust to failures of single system components and enable graceful degradation. Behavior based systems enable modular system development and this support the trends to choose a decentralized control scheme, but distributed systems add the overhead of coordination, synchronization and communication between system components, which must be overcome by powerful software engineering tools.

Important issues are still facing behavior-based systems and need to have proper solutions, such as, behavior generation, how to design simple behaviors that guarantee robust operation in spite of the limited knowledge available at design time, behaviors and implicit representation, behaviors and experience, distributed approach in behavior control, uncertainty handing that reflects the problem of extracting reliable evidence from unreliable sources of information, multi-valued actions and action selection, and how to coordinate the activity of several and possibly competing behaviors in order to perform a complex task, etc. In the behavior-based approach the control of a robot is shared between multiple behaviors with different and possibly

incommensurable objectives. In most cases when deciding what next action to take, multiple conflicting objectives should therefore be considered simultaneously. This is known as the action selection problem.

4. Action Selection Mechanisms

Autonomous intelligent systems are characterized by the fact that they select one from a set of equivalent action alternatives in a given situation as appropriate. Most of the existing action selection approaches that have been implemented on a real mobile robots are hard-wired have been carried out with fixed behavior outputs. To make these systems more adaptable to various situations and goals to pursue in the world, it is necessary to dynamically select behaviors and to change their respective priority to make the system behave appropriately according to the situations it encounters in the real world. External and internal variables like motivations can also be used to affect the selection of behaviors. In addition, such intelligent and autonomous systems can learn from their experiences in different ways to improve their capabilities [11]. They can choose completely different actions if they happen to be confronted later with a similar situation. This learning can be achieved differently: by adapting some parameters, acquiring new knowledge of an application field, or by evolutionary processes from which generations of systems can be developed to approximate a specific fitness function. An intelligent robot should adapt its emerging behavior to its perception of the environment, its needs, its knowledge and its ability to satisfy its intentions

Behaviors typically produce a single-valued output that is viewed as best to perform an action. But, at some input states of the environment, several behaviors may become active and there may be multiple actions competing for execution. The problem of run-time choice between multiple, parallel, competing, conflicting and overlapped goals demand that the system must respond to unpredictable and passing opportunities in the world where the robot is moving. This will tell how does a mobile robot decides which goal to pursue at a given moment, when to interrupt, and when to opportunistically divert in response to events within its environment. Thus a major issue in the design of systems for control of autonomous mobile robots is the formulation of effective mechanisms for coordination of the behaviors activities into strategies for rational and coherent behavior. However, due to several constraints such as environmental complexity and unpredictability and due to robot's limited resources, action selection cannot be completely rational or optimal. Thus, it is appropriate to consider selecting good enough actions that satisfy the objectives.

Behaviors can be categorized into two categories: homogeneous behaviors that include behaviors with common or similar objectives, and heterogeneous behaviors that include behaviors with distinct objectives. Action selection for homogeneous behaviors is easier since the behaviors have the same objective and should ideally agree on what action or set of actions needs to be selected. Hence, it is the case that involves the selection of an action that is optimal with respect to all objectives. Homogeneous behaviors imply redundancy, and redundancy can be exploited to improve the reliability of the system and to enable uncertainty handling. In contrast, action selection for heterogeneous behaviors is more complicated since each behavior has different objectives; thus conflict/contradiction occurs and optimal solution might not exist and hence it is important to have a mean to resolve conflict situations, decide and select the most appropriate action from a set of competing actions.

Action selection has been studied from several different points of view. However, the main research direction can be categorized into the following fields: Ethology, Artificial life, Virtual reality and graphical simulation, software agent, and physical robots. There are a number of action selection mechanisms mainly for control of robots and some for control of synthetic creatures, these can be divided into classes of action selection mechanisms. Among the available mechanisms two groups can be classified; state based (arbitration) and continuous (command fusion) [12, 14].

The arbitration is applied when a set of robot's behavior are relevant in a given context. It allows one or a set of behaviors at a time to take control for a period of time until another behavior or a set of behaviors is activated. Only one action controls the system at a time, hence no fusion of outputs is needed. The selection of the next action is either done in a distributed way or by a higher-level component [15, 19]. Advantages of this category are reflected mainly by the ability of behavior sequencing and the effective use of resources, but it has no support for cooperative control. Some of the approaches that are classified under this category are:

- Priority-based. The example for this group is the Subsumption architecture [15], where behaviors with higher priorities are allowed to subsume the output of behaviors with lower priority and hence be eligible to take control of the robot. Most of the early behavior-based approaches on action selection employed hard-wired mechanisms, which involve pre-fixing the behavior outputs at design time. Real world implementations on mobile robots have been carried out with fixed behavior outputs. However, difficulties exist with hard-wired schemes on real environments, i.e., unable to cope with unpredictable real environment.

- State based arbitration. In this framework the interaction of behaviors are formalized by the Discrete Event System (DES) formalism in a modular and hierarchical fashion [16]. The behavior selection mechanism in this approach is implemented using state transition, where states correspond to execution of actions/behaviors and events, correspond to observations and actions, cause transitions between the states. Upon detection

of a certain event a shift is made to a new state and thus a new behavior. The temporal sequencing approach also known as perceptual sequencing and sequenced coordination [17] is very similar to the DES approach. A finite-state automaton is used for formulating a mechanism for sequencing between a series of behaviors based on perceptual triggers. At each state a distinct behavior is activated and perceptual triggers cause transitions from one state to another. The advantage of this approach is recognized because it is based on the finite-state machines formalism and thus system observeability and controllability can be shown for a particular implementation. The major drawback of this approach is that this modeling scheme is known to be an NP-hard problem.

Bayesian Decision Analysis [18] addresses the related problem of sensor selection. Sensor selection can be considered a special case of action selection where the actions are certain sensor operations. A fundamentally different approach to action selection is to learn the action selection mechanism. Several approaches to learning of action selection exist and the most promising among these techniques is the reinforcement learning [14].

Winner-take-all. An example is the activation network approach, where no central module is required [19]. In this approach the system consists of a set of behaviors or competence modules that are connected to form a network. Action selection results from the interaction of a set of distributed behaviors that compete until one behavior wins the competition and takes control of the robot. Evidentially this action selection mechanisms deals only with selection of behaviors and not with motor actions. When a behavior is selected it will perform the most appropriate action from its point of view. This particular approach has been applied to simulated problems and has not been evaluated on real-world scenarios, which could help discover its strengths and limitations. One shortcoming of this approach might be that the state of the environment does not include any uncertainty thus the system assumes perfect sensors. However, due to its distributed nature the system is fault-tolerant since the failure of a single behavior does not necessarily lead to the failure of the entire system [14].

The command fusion category is used to coordinate activities of the set of behaviors that are active simultaneously in a given context. To ensure a coherent behavior, command-fusion or continuous coordination mechanisms must coordinate activities of the set behaviors that are active simultaneously and that are related to one another. The advantages of this category are supporting and facilitating cooperative control, blending behaviors, and handling multiple objectives, but it has no support for sequencing. The Approaches that can be classified under this category are:

- Voting. The example for this group is the DAMN architecture [20]. DAMN is a distributed architecture for mobile navigation that consists of a set of behaviors to pursue the system goals based on the current state of robot's environment. This technique interprets the output of each behavior as votes for or against possible actions and the action with the maximum weighted sum of votes is selected. A behavior can be a reactive behavior as well as a planning module. These weights reflect the relative importance or priority of the behavior in a given context. However, there is no description on how the mode manager determines the weights. A variety of voting schemes can be used such as, unanimity voting, majority voting, M-out-of-N voting; plurality voting that selects action with maximum number of votes, etc. It has been shown that plurality voting has a higher probability of choosing the correct action when compared to a number of other voting schemes, under the assumption of statistical-independence

- Artificial neural network architecture is proposed where the emphasis is placed on making the action selection mechanism adaptive to short and long-term environmental and goal changes. This will facilitate an adaptive estimation of each behaviors contribution to the final result.

  Also, fuzzy inference techniques are used to formalize the voting approach by establishing the relation between multiple objective action selection and fuzzy inferencing in the context of action selection.

- Superposition approaches. This approach combines behavior recommendations using linear combinations. It includes potential-fields, motor schemas [21] and the dynamic systems approach.

In most systems the use of both mechanisms is necessary because they are not competitive but complementary. Additional factor can be used in different ways to affect the selection of behaviors. This factor is the observation of use of behaviors and their effects. It gives indirect information on the interactions between the system and its environment. Another approach uses behavior exploitation to affect the selection of behaviors to decrease the burden of the designer in knowing everything about the environment. The fundamental idea is that a purpose (applicability conditions) is associated with each behavior, and this can be used to reason about the way an intelligent robot is behaving in its environment.

5. Conclusions

The presented work in this paper shows that it is important for the knowledge capacity of an intelligent robot to adapt to make it possible for the robot when losing certain sensory capabilities or due to internal and external uncertainty to navigate safely and purposely

throughout their environments. Also, it is important to recognize that designing effective action selection mechanisms is a major endeavor in the design of an intelligent and autonomous robot. A related problem to the action selection is the reliability of the decisions the robot makes. It is clear that if each behavior generates a multi-valued output a more moderate composition scheme could be constructed. Multi-valued outputs seem to make an appropriate basis for communication among the behaviors, better means of compromise and to reduce the amount of loss of information. Parallel, abstract, distributed, and active representation is essential, in order to accommodate real-time demands of the system.

Since behavior modules take part at different levels of the control hierarchy, an efficient action selection mechanism should be devised to deal with scheduling, management, coordination and communication between modules constituting behavior based systems so that coherent behavior can be achieved. Learning to select appropriate actions is still an open challenge in terms of time it is required, complexity of task and the environment.

References

[1] E. Gat, Towards Principled Experimental Study of Autonomous Mobile Robot, In proceedings of the International Symposium of Experimental Robotics, Stanford CA, 1995.

[2] U. Nehmzow, T. Matsui, and H. Asoh, "Virtual Coordinates: Perception-Based Localisation and Spatial Reasoning in Mobile Robots", Proceeding of the International Conference on Intelligent Autonomous Systems 5, Japan, 1998.

[3] J. Yen, and N. Peluger, "A Fuzzy Logic Based Extension to Payton and Rosenblat's Command Fusion Method for Mobile Robot Navigation", IEEE Trans. on Systems, Man and Cybernetics, Vol. 25, No. 6, pp. 971-978, 1995.

[4] A. G. Skarmeta, and H. M. Barberá, "Fuzzy Logic Based Intelligent Agents for Reactive Navigation in Autonomous Systems", 5th International Conference on Fuzzy Theory and Technology (FT&T'97), Durham, USA, pp. 168-171, 1997.

[5] R. Cattoni, "Behaviours as Bridges Between Symbolic Reasoning and Reactivity", In Proc. of the Italian Workshop on Planning IPW'93, Italy, 1993.

[6] J. Van-Dam, "Environment Modeling for Mobile Robots: Neural Learning for Sensor Fusion", Ph. D. Thesis, University of Amsterdam, 1998.

[7] S. Lacroix, S. Flury, H. Haddad et al, "Reactive Navigation in Outdoor Environments", International Journal of Robotics Research, 1998.

[8] U., Zimmer, "Autonomous Physical Systems", First International NAISO Congress on Autonomous Intelligent System, Australia, 2002.

[9] M. Nicolescu, and M. Mataric, "A Hierarchical Architecture for Behavior-Based Robots", Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems, Bologna, Italy, 2002

[10] M. Nicolescu, M. Mataric, "Deriving and Using Abstract Representation in Behavior-Based Systems", Proceedings, Seventeenth National Conference on Artificial Intelligence (AAAI), Student poster, page 1087, Austin, Texas, 2000.

[11] F. Michaud, "Adaptability by Behavior Selection and Observation for Mobile Robots", In book, Soft Computing in Engineering Design and Manufacturing, R. Roy, P. Chawdhry and P. Pants (eds), Springer-Verlag. 1997.

[12] D. C. Makenzie, R C Arkin, J M Cameron, Specification and Execution of Multiagent Missions, Autonomous Robots, Vol. 4, No. 1, 1997.

[13] P. Maes, "Designing autonomous agents", in P. Meas (ed.) Designing autonomous agents, Cambridge, MA: MIT Press, 1990.

[14] P. Pirjanian, " Behavior coordination mechanisms - state-of-the-art", Tech Report IRIS-99-375, Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, California, 1999.

[15] R. A. Brooks, A Robust Layered Control System for a Mobile Robot, IEEE Journal of Robotics and Automation, Vol. 2, No. 1, pp. 14-23, 1986.

[16] J., Kosecka and R., Bajcsy, "Discrete Event Systems for Autonomous Mobile Agents", Proceedings of the Intelligent Robotic Systems '93, pp. 21-31, 1993.

[17] R. C., Arkin, and D., MacKenzie, "Temporal Coordination of Perceptual Algorithms for Mobile Robot Navigation", IEEE Transactions on Robotics and Automation, 10(3), 276-286, 1994.

[18] S. Kristensen, "Sensor Planning with Bayesian Decision Analysis", PhD thesis, Department of Medical Informatics and Image Analysis, Aalborg University, 1996.

[19] P., Maes, "How To Do the Right Thing", Technical Report NE 43-836, AI Laboratory, Massachusetts Institute of Technology, 545 Technology Square, Cam-bridge, MA 02139, USA, 1989.

[20] J. K., Rosenblatt, "DAMN: A Distributed Architecture for Mobile Navigation", in AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents, Stanford, CA-USA, 1995.

[21] R. C. Arkin, "Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior", IEEE ICRA'87, pp. 264-271, 1987.

[22] A. Saffiotti, "Some Notes on the Integration of Planning and Reactivity in Autonomous Mobile Robots", AAAI'93, pp. 122-126, 1993.

[23] M. Mataric, "Behavior-Based Control: Examples from Navigation, Learning and Group Behavior", Journal of Experimental and Theoretical Artificial Intelligence, Vol. 9, No. 2-3, pp. 323-336, 1997.

[24] R. J., Firby and M., Slak, "Task Execution: Interfacing to Reactive Skill Network", AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures fo Physical Agents, pp. 92-96, 1995.

Explore the gait stability of a biped robot prototype based on the finite element analysis

Jiwu Wang

Department of Precision Instruments and Mechanology,
Tsinghua University, Beijing 100084, P.R.China
(Email: wang_jiwu@hotmail.com)

Xing Ouyang   Ken Chen

Department of Precision Instruments and Mechanology,
Tsinghua University, Beijing 100084, P.R.China

Abstract

Achieving walking balance is necessary for a biped robot, no matter static walking or dynamic walking. Due to the multi-rigid body, nonlinear, many DoFs and the open loop structure in single support phase and close loop structure in double support phase, it is difficult to prove its stability for a dynamic walking with present control method, especially when the impact on the floor and vibration were considered. The COM and ZMP are present tools to control balance while walking. But they can only be calculated approximately. In this paper, the finite element analysis method was introduced to explore the gait stability for a biped robot walking. The COM and ZMP were calculated separately for each instance in a gait cycle.

1. Introduction

How to keep balance is important for a biped robot when it carries on a static walking or dynamic walking. There are many factors that will destroy the balance, such as the bad distribution of degrees of freedom (DOFs), the clearance of installation and fabrication tolerance, the vibration caused by walking, the impact made by a foot stricken on the floor and the bad gait design etc. Some factors can not be avoided. For the static walking, a system will stay balance by always keeping the center of mass (COM) of the system vertically projected over the polygon of support formed by the feet. While the dynamic walking means the center of gravity of the robot is not projected vertically onto the area of either foot by a small period in the walking cycle. That is, there is a period of controlled instability in the gait cycle. But the COM in s static walking must not leave the area of support formed by the feet still in contact with the ground whenever a foot or leg is moved. In order to keep balance for a dynamic walking, Zero Moment Point (ZMP) is calculated. Therefore for a dynamically stable walking, the ZMP must locate in the polygon of support formed by the feet, but the COM may leave the support area formed by the feet for periods of time. So dynamic walking is more efficient than static walking. But it is difficult to control balance in a dynamic walking.

In the view of above analysis, regardless of static walking or dynamic walking, in order to keep balance, it is necessary to calculate the COM or ZMP at every instance in a step period. Till now, it is nearly impossible to prove the stability of dynamically walking two-legged robots using the present control techniques. The effective solutions were how to compensate such defects. More efforts were put on how to design a more reliable and robust gait which can keeps balance under different ground conditions. And more researchers paid more attention on the control strategies. Their effect will be verified with various purposive experiments. It generally will take a long time to design such experiment and prepare necessary instruments and sensors. Moreover, some information can not be gotten directly due to the limits of present technical conditions. In this paper the finite element model of our biped robot prototype was built to analyze the balance of different gaits. In order to

get reliable results of simulation, we try to build the finite element model as the same as the biped prototype. In the process of simulation, we will separate the whole gait in each gait period into some independent postures which include these key instances to keep a balance. Based on the simulation, the COM and ZMP for each instance can be calculated, the vibration in walking will be analyzed and the foot impact of the swing leg on the floor will also be calculated.
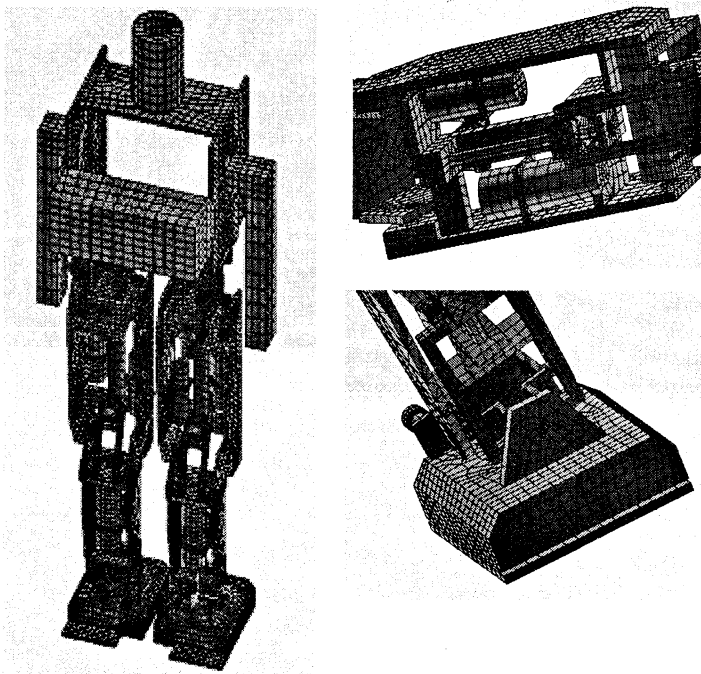
Fig.1 The 3D finite element model of the biped robot prototype

The three dimensional finite element model of our prototype has been built, as shown in figure 1. All components were meshed with hexahedral or tetrahedral elements. The angle speed and acceleration in each instance calculated by gait analysis for controlling the prototype will be added to the corresponding joints of the finite element model as loads. Thus the COM and ZMP will be calculated. The gait will be tested if it can make the robot keep walking balance.

2. Conclusions

The three dimension finite element analysis method will be applied to the study of the walking balance of out biped robot. The three dimension finite element model has been built. And the further analysis will be carried on in the following research.

References

[1] Elliot Nicholls, Bipedal dynamic walking in robotics, Thesis of the University of Western Australia, department of electrical and electronic engineering, 1998

[2] Allen S.Parseghian, Control of a simulated, three-dimensional bipedal robot to initiate walking, continue walking, rock side-to-side, and balance, Thesis of Massachusette institute of technology, 2000

[3] Paul W.Lathyam, A simulation of bipedal walking robots: modeling, walking algorithm, and neural network control, Thesis of B.S.E.E. University of Minnesota, 1980

[4] Daniel Joseph Paluska, Design of a humanoid for walking research, Thesis of Massachusetts Institute of Technology, 2000

[5] Jerry E.Pratt, Gill A.Pratt, Exploiting natural dynamics in the control of a 3D bipedal walking simulation, International conference on climbing and walking robots (CLAWR99), Portsmouth, UK,1999

[6] Jerry E.Pratt, Exploiting inherent robustness and natural dynamics in the control of a bipedal walking robots, Thesis of Massachusetts Institute of Technology, 2000

[7] Yasutaka Fujimoto, Satoshi Obata, Atsuo Kawamura, Robust biped walking with active interaction control between foot and ground, Proc. Of the IEEE int. conf. On robotics & automation, Leuven, Belgium, 1998

Development of a dynamically stable gait for a biped robot prototype

Jiwu Wang

Department of Precision Instruments and Mechanology,

Tsinghua University, Beijing 100084, P.R.China

(Email: wang_jiwu@hotmail.com)

Zhao Jiandong   Ken Chen   Lijun Shao

Department of Precision Instruments and Mechanology,

Tsinghua University, Beijing 100084, P.R.China

Abstract

Walking balance, an important characteristic for a biped robot, is difficult to kept in dynamic walking. Although more control strategies were given till now, the center of mass (COM_) and zero moment point (ZMP) are two important targets that have to be considered for a gait design. In this paper, considering the control parameters: step length, step height and step period, a gait for dynamic walking was developed using periodic function approach. And the COM of the robot at every instance were calculated in such a way that coupling between joints are taken into account. Each joint is servoed independent from the others. Therefore, it is convenient to control the biped robot to stabilize a dynamic gait, especially, to vary the speed and position of the robot.

1. Introduction

Walks are complex activities. When a person is beginning to walk, he not only raises one foot and moves forward, but also moves his hips, knees, arms, waist and head etc at the same time to maintain balance. For a biped robot, it is required to design a reliable gait before it can walk steadily. A gait cycle can be simply divided into a double support phase and a single support or swing phase. In the period of double support, both feet are in solid contact with the ground. It starts when the front foot begins to contact with the ground, and ends when the rear foot completely breaks contact with the ground. If it carries on a static walking, the weight of its body moves from the rear foot to the front foot in this period. In the remainder part of a gait cycle, one foot keeps solid contact with the ground, and another foot swings from the rear to the front. For a dynamic walking, in the first half cycle of the swing phase, the weight of its body is transferred to the front foot, and the knee bends to absorb the shock. Then the second half of the swing phase starts: the support knee straightens out to uphold its body and swings forward like an inverted pendulum until another foot beginning to contact with the ground. The walking pattern alternates the single support phases between each leg, and interspersed with a double support phase between each alternation. Till now we have no time to consider the movement of the arms and head.

For the same distance, dynamic walking is more efficient than static walking because the double phase of former is shorter than the latter[1]. Here dynamic walking is characterized by a small period in the walking cycle where the center of gravity of the robot is not projected vertically onto the area of either foot. This requires there to be a period of controlled instability in the gait cycle. Static balance or static walking refers to a system which stays balance by always keeping the center of mass (COM) of the system vertically projected over the polygon of support formed by the feet. While this is the case there can be no horizontal acceleration due to tipping moment caused by gravity. Therefore whenever a foot or leg is moved, the COM must not leave the area of support formed by the feet still in contact with the ground. But for dynamic walking, the COM may leave the support area formed by the feet for periods of time. This allows the system experience tipping moments, which give rise to horizontal acceleration. However, such periods of time are kept brief and strictly controlled so that the system does not become unstable. In order to achieve balance while walking, it is necessary to develop a dynamic gait or walking pattern first. In this paper, considering the control

parameters: step length, step height and step period, a gait for dynamic walking was developed using periodic function approach. Therefore, it is convenient to control the biped robot to stabilize the dynamic gait, especially, to vary the speed and position of the robot.

On the other hand, it is nearly impossible to prove the stability of dynamically walking two-legged robots using the present control techniques[2]. This is because such system is highly nonlinear and naturally unstable so that legged motion researchers are yet to come up with a convincing, mathematically based control system that can fully explain why a biped is able to walk or fail to walk continuously. At the same time, bipeds are multi-input, multi-output systems that are both continuous and discrete. While in single support, the system operates in a continuous fashion, as soon as the support leg switches, there is discreteness, as well.

In this paper, in order to realize dynamically stable walking, the position of the center of the mass and center of pressure of the robot at every instance were calculated in such a way that coupling between joints are taken into account. Each joint is servoed independent from the others therefore making control more intuitive. Position of the center of the mass and center of pressure of the robot are controlled using ankles, therefore every time the couplings between all the joint of the robot are taken into account. The control method is very simple to understand, and it does not require dynamic calculations of the robot. The Zero Moment Point (ZMP) can be calculated with six-axis force sensors installed in each foot.

In the following part, we will give the method to develop a gait with a periodic function approach. At the same time, we will also discuss how to control with six-axis force sensors.

2. The periodic function approach

There are many approaches to develop a gait for a biped robot. Here we mainly discuss a method using periodic function to design a gait. Firstly, we assume the arms do not move and they keep a whole with its body. Since walking is a repetitive motion which repeats over time, it is possible to design a gait based on the relationship of each joint angle over time. Here we do not consider the influence of vibration and foot impact etc. In order to alter the walking velocity and scale the gait conveniently for different requirements, three parameters: the step length, step height and step period were considered to develop a gait. For a gait, if we can specify the foot trajectory (mainly the swing foot, the support always keeps contact with the ground in the period of single support phase) and the above three parameters, the joint angle of each link will be solved based on the inverse kinematics.

Here we take the position of the support foot in the first step as the origin for the coordinate system. The position of the swing foot can be expressed as:

$$x_f = \frac{4L_{step}t}{T} - L_{step}$$

$$y_f = H_{step}\sin(\frac{2\pi t}{T})\sin\theta_{yaw}$$

$$z_f = H_{step}\sin(\frac{2\pi t}{T})\cos\theta_{yaw}$$

Where x_f represent forward direction, y_f represents side direction and z_f represents height direction in the three dimensional coordinate system. L_{step} is the step lemgth, H_{step} is the step height and T is the step period. θ_{yaw} is the angle of yaw and t is the time.

At the same time, based on the trace of swing foot, we can also get the position of the swing hip as follows:

$$x_{hip} = (L_{Uleg}\cos\theta_{pitch} + L_{Lleg}\cos\theta_{pitch})\sin\theta_{hip}$$

$$y_{hip} = (L_{Uleg}\cos\theta_{pitch} + L_{Lleg}\cos\theta_{pitch})\sin\theta_{hip}\sin\theta_{yaw}$$

$$z_{hip} = (L_{Uleg}\cos\theta_{pitch} + L_{Lleg}\cos\theta_{pitch})\sin\theta_{hip}\cos\theta_{yaw}$$

Where x_{hip}, y_{hip} and z_{hip} represents the position of the swing hip in the three dimensional coordinate respectively. L_{Uleg} is the length of the upper leg and L_{Lleg} is length of the lower leg. θ_{pitch} is the pitch angle to keep the position of COM before walk. θ_{hip} is the angle of the swing hip joint in walking and θ_{yaw} is yaw angle of the whole body in walking.

Then, based on the Law of Cosines, we can get the knee angle easily.

Thus we can get the trace for each joint in the swing leg. Because the support leg keeps in touch with the ground in the whole swing phase, for simplification, we can assume the joints in the support leg were locked at this moment. In figure 1 we give the results for a gait of 6 second with step length of 30cm and step height of 5cm. Due to the periodic characteristic, we only calculate the angles for each joint of three representative parts: the start walking, one whole step from the start of a single foot support phase to the end of a double feet support phase, the end walking.
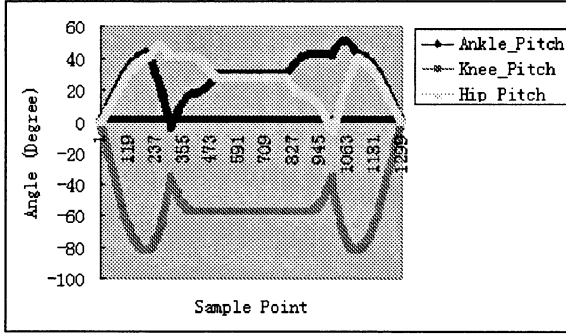
a) Left leg
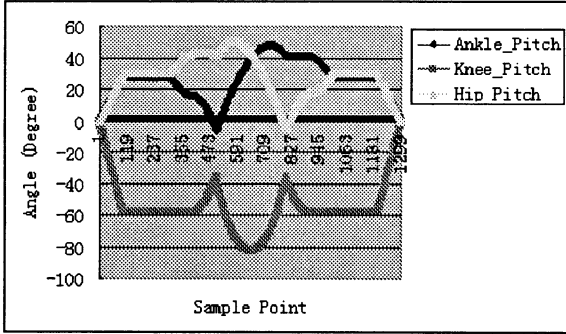
b) Right Leg

Figure 1 The illustration of the joint angle for each leg in three representative walking state

At the same time, the corresponding angle speed of each joint was also calculated. They were shown in figure 2.
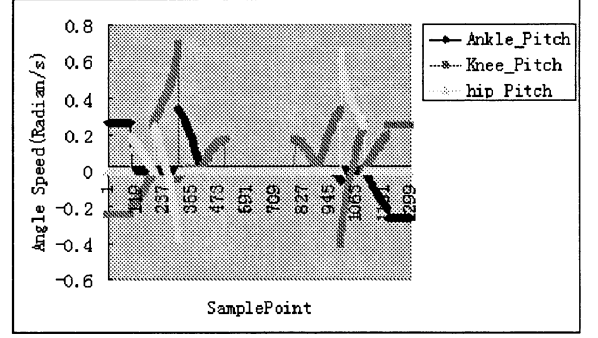
a) Left leg
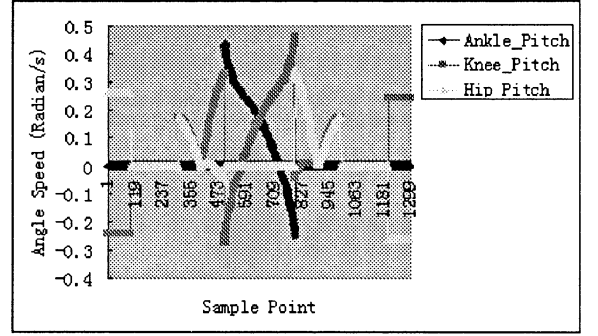
b) Right leg

Figure 2 The illustration of the joint angle speed for each leg in three representative walking state

3. The Calculation of ZMP

Based on trace of the ankle and hip, we can calculate the joint angle and speed for each joint with the periodic function method. But we are not sure that gait can keep the robot balance while walking. In order to make the robot realize stable walking, the COM and ZMP must be calculated for each instance in a gait period. For our biped robot, we mainly use the six-axis force sensor to determine the ZMP. The ZMP can be calculated as follows:

$$
\begin{cases}
(y_s - y_P)F_z - z_s F_y + (y_a - y_P)m(\ddot{z}_a - g) - z_p m\ddot{y}_a + M_x + M_{Px} = 0 \\
z_s F_x - (x_s - x_P)F_z + z_p m\ddot{x}_a - (x_a - x_P)m(\ddot{z}_a - g) + M_y + M_{Py} = 0 \\
-(y_s - y_P)F_x - (x_s - x_P)F_y - (y_a - y_P)m\ddot{x}_a + (x_a - x_P)m\ddot{y}_a + M_z + M_{Pz} = 0
\end{cases}
$$

Where, M_{px}, M_{py} and M_{pz} is the moment at the ZMP respectively. M_x, M_y and M_z is the moment acted on the sole respectively. F_x, F_y and F_z is the reaction force acted on the sole respectively except the effect of inertia.

3. Conclusions

Considering the parameters of step length, step height and step period, we can design various gait with periodic function method. These parameters are also important to control balance while walking. Because the complexity of ground conditions in practical applications, such as obstacle avoidance, climbing stairs, this method is useful for such cases.

References

[1] Elliot Nicholls, Bipedal dynamic walking in robotics, Thesis of the University of Western Australia, department of electrical and electronic engineering, 1998

[2] Allen S.Parseghian, Control of a simulated, three-dimensional bipedal robot to initiate walking, continue walking, rock side-to-side, and balance, Thesis of Massachusette institute of technology, 2000

[3] Paul W.Lathyam, A simulation of bipedal walking robots: modeling, walking algorithm, and neural network control, Thesis of B.S.E.E. University of Minnesota, 1980

[4] Daniel Joseph Paluska, Design of a humanoid for walking research, Thesis of Massachusetts Institute of Technology, 2000

[5] Jerry E.Pratt, Gill A.Pratt, Exploiting natural dynamics in the control of a 3D bipedal walking simulation, International conference on climbing and walking robots (CLAWR99), Portsmouth, UK,1999

[6] Jerry E.Pratt, Exploiting inherent robustness and natural dynamics in the control of a bipedal walking robots, Thesis of Massachusetts Institute of Technology, 2000

[7] Yasutaka Fujimoto, Satoshi Obata, Atsuo Kawamura, Robust biped walking with active interaction control between foot and ground, Proc. Of the IEEE int. conf. On robotics & automation, Leuven, Belgium, 1998

Study on Humanoid Robot Joint Servo Control System Based on Can-Bus

SHAO Li-jun
Department of
Precision Instruments and Mechanology
Tsinghua Universityeijing, China. 100084
Shanny@tsinghua.org.cn

ZHAO Jian-dong,
WANG Ji-wu,
WANG Jin-song,
CHEN Ken

Abstract

In the view of the special requirements of the structure and control system of the biped humanoid robot prototype, this paper presented a project of the hierarchy servo control system based on CAN-bus for the biped robot THBIP-I. The author particularly describes the systemic project, structure of servo card, communication protocol and control algorithm, and realization of prototype's system. It has been proved to be an effective and feasible control system by practical application.

Keywords

Humanoid robot, Field Control Bus, CAN-Bus, Position Servo

1 Introduction

Humanoid robot is a popular research in the field of science and technology now, but there are still many theoretical and technical questions to be solved during the development, joint servo control technology is such a key problem.

THBIP-I, the first model made by humanoid robot research team in Tsinghua University [1], is a multi-input and multi-output, high coupling, nonlinear and complex multi-rigid body biped walking dynamic system. Our servo control system should have the following features demanded by the characters above.

1) High integration, small volume. The whole system can be installed in the robot. There should be fewer amounts of hardware and wires as possible, and the size should be more succinct.

2) High servo behavior. It can fit complex dynamic control requests, such as variant parameters, high coupling, and gravity effect. The system should have fast dynamic response, and lesser steady-state error.

3) High reliability. When any unit is out of work itself or has some trouble in communication, it can alarm and auto-protect in time, in the meanwhile the other units can't be influenced.

4) Strong anti-jamming ability. The control system can work normally in the electromagnetic interference, which is generated by running motor.

Therefore, in order to control THBIP-I efficiently, this paper studied on the joint servo control technology and system design about the humanoid biped robot.

2 Structure of servo control system

With the development of computer, control, information, network and microelectronics, field-bus control system, the opening, digital, decentral and intelligent bottom control network, which is constituted by field-bus based on bi-direction, serial, multi-node communication, has been used in various control fields successfully now. The basic control unit gradually integrates current-loop, velocity-loop, position-loop control, forming entire digital motor servo driver of high performance. Based on the principle of downloading field-bus control function, microprocessor and field-bus interface were put into entire digital motor servo driver, thus intelligent and independent digital control executive unit come into being. The control system made of this unit has many prominent advantages on veracity, real-time, patency and reliability.

There are many kinds of field-bus and intelligent drivers, but their large scale or high price make it difficult to fit our servo control request. According to these requirements, we presented a project of the hierarchy servo control

system for the biped robot based on CAN-bus, combining with the system scale, working environments, requirement of the signal transmission and the local equipment conditions.
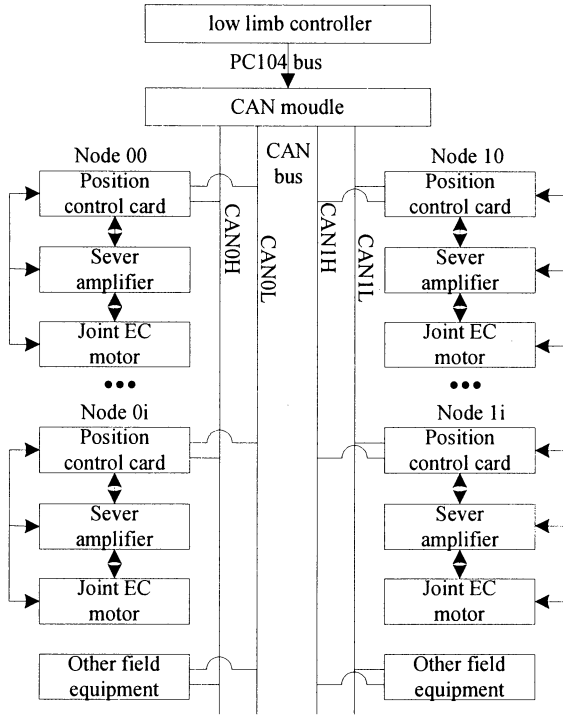
Fig.1 Configuration of control system based on CAN bus

As Fig. 1 shows, The PC104 CPU-module is used as the master computer in the system; it can control dual port CAN-bus interface module by PC104-bus. As a master-node, each CAN-bus port communicates with 6 joint-nodes, which includes the position control card designed by ourselves with CAN-bus port, the digital servo-driver integrating current-loop and velocity-loop, and DC brushless servomotor. Thereby the whole servo control system forms two independent CAN-bus networks.

After finishing mission programming, master transmits command message to every node (position control card, other field equipment) by dual port CAN-bus interface module according to communication protocol. Every node decipher command, then make corresponding flow. When it receives motion command, the control card gathers current joint position and returns it to master, then executes control algorithm, outputs command voltage to servo driver to drive joint motor. Other

normal nodes do this in the same way, and communicate with master and other slaves to share data.

3 Design of joint position servo card

The Controller Area Network (CAN) is a serial communications protocol, which efficiently supports distributed real-time control with a very high level of security. It's the only field-bus authorized as the international standard by now. Its domain of application ranges from high speed networks to low cost multiplex wiring. In automotive electronics, engine control units, sensors, anti-skid-systems, etc [2].

Based on the well features of CAN, this paper designed a position control card shown in Fig. 1. The card uses Intel 16-bit MCU 80C196KC, which expands 32K program and data storage space. The configuration of position control card is shown in Fig.2. According to the input mode of servo driver
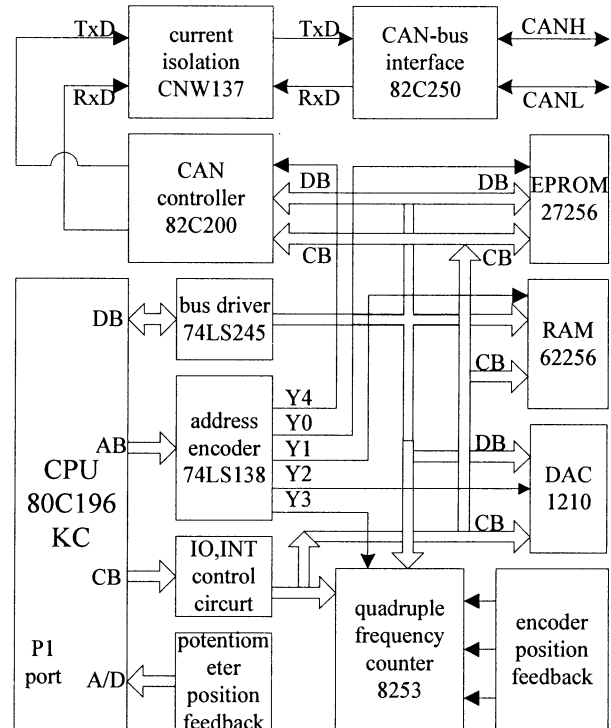
Fig.2 The configuration of position control card

command and the scale of joint velocity, we select 12-bit parallel DAC1210 as DA conversion chip, and design follower to improve anti-jamming ability; considering full-loop and half-loop, we designed two kinds of joint position feedback:

encoder feedback and potentiometer feedback, encoder signal. After being difference driven, the encoder signal is disposed with fourfold frequency in order to count in positive and negative direction. A 10-bit AD converter of a MCU samples the potentiometer signal. Port 1 is used as I/O port, for controlling the brake and enable signal. We use stand-alone CAN controller 82C200 produced by Philips as CAN-bus interface. Combined with photo electricity isolation, the drive bus interface (82C250) has been selected, in order to support the function of difference sending and receiving.

4 Communication protocol and control algorithm

Master software presides over mission planning and the globe controlling, it can be divided into three modules according to function: initialization, communication and display. The second is the most important, which can transfer control data to
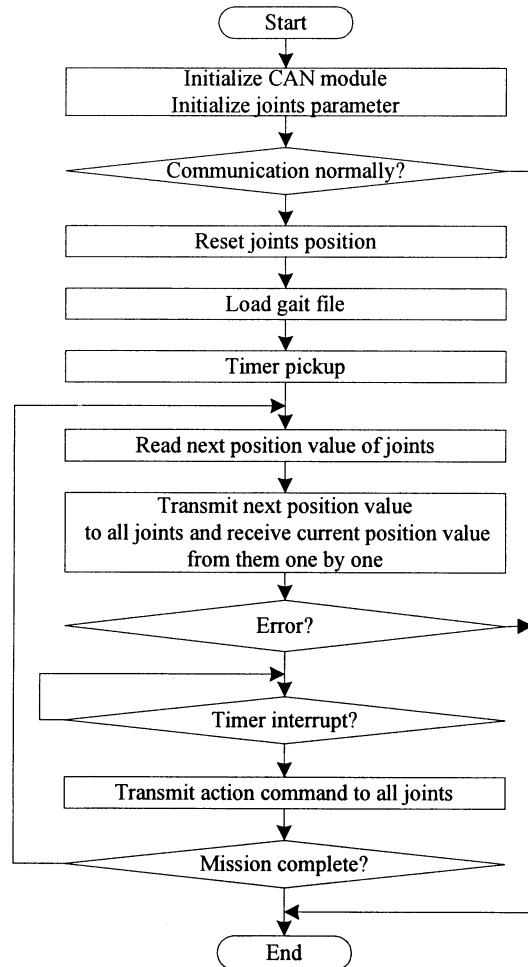
Fig.3 The main flowchart of master machine

joint-nodes through communication prototype and analyze current status by the feedback message, then make corresponding actions. The main program flowchart is shown in Fig.3. Since two CAN masters transferring and receiving alternately is used, the communication time is shortened efficiently. A round of communication with 12 nodes can be completed within 1.2ms, which makes it possible to shorten control cycle to 5ms. At the same time, it can insure execution of two CAN-networks synchronistically by using transfer executive frame simultaneously. The maximal time difference between two nodes is several CPU cycles, about 10ns. Apparently, it should be ignored in the control cycle (See Fig.4).

T = Transmit message to node ...
R = Receive message from node ...
TA = Transmit action command to all nodes
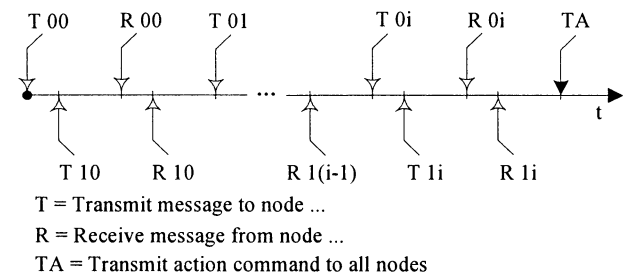
Fig.4 The communication order of nodes with master

The software of position control card consists of monitor, communication and control algorithm. When the biped robot walks, every joint-node can get next target position from master and return current position, then execute corresponding control algorithm by protocol, send voltage command to servo-driver to drive joint-motor. The

Table 1: Communication protocol of control system

Flag	Action description
0	Abort. Ask all joints to abort action, disable the motor and enable its break.
1	Get current joint position value from slave.
2	Communication test. Transmit a random message and wait for return.
3	Disable the break of the motor and enable the motor of the joint.
4	Reset. Reset all joints to original position.
5	Transmit target position value to one joint and get current position value from it.
6	Execution. Run motion as setting.

ID.2~ID.0 of communication message are used as the action flag for communication protocol, as Table 1 presents. In order to insure reliability of control system, Author programmed faultless error handling module by cyclically checking position, velocity, communication, DA and AD, etc.

These programs have some agility in adjusting control parameters. Because the controlling ability of each node varies, algorithm selection depends on the actual condition. For example, we have used simple PI and dual lag inertia loops (See Fig.5), etc.

5   Experimental results

The effectiveness of the joint servo control system was examined by experiments with humanoid robot, THBIP-I. One view of the robot is shown in Fig.5. All sensors, motors, drivers, cards and computers have been installed in the body and tested.
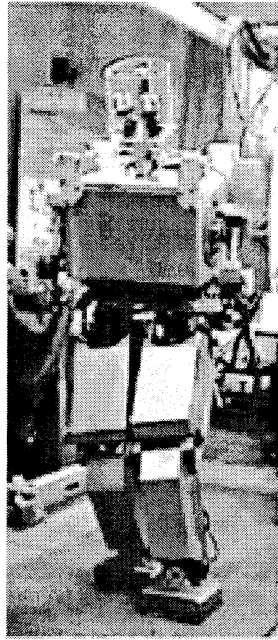
Because the controlling ability of each node varies, algorithm selection depends on the actual condition. Fig.6 shows the motional curve of robot's left ankle joint with P adjust and DLIL (Dual Lag Inertia Loops) adjuster (See Fig.7). Apparently, DLIL adjuster is much better than P adjuster on system response.

Fig.5 THBIP-I
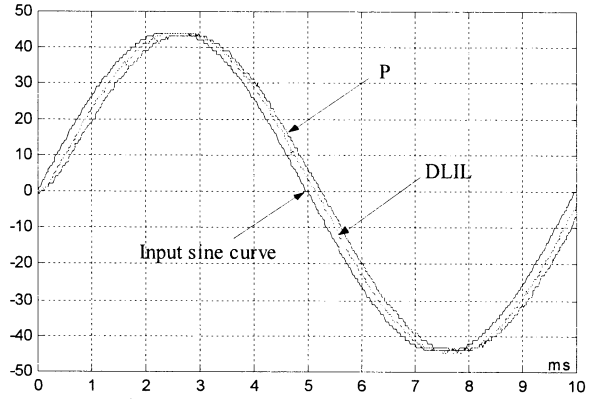
Fig.6 The comparison of two adjusters

Based on this servo corol system, our THBIP-I accomplished static walking continuously by its own master. With 20ms control cycle, his maximum speed is 8s/step with 300mm stride [1], and it will be improved by reducing control cycle in future.

6   Conclusions

The control system based on CAN-bus and the position control card have been debugged successfully on THBIP-I, the basic gait controlling can be accomplished. The master and slave work correctly and reliably, the system has a nice expansibility, thus it is an effective and feasible control system.

Reference

[1] Kai Xu, Ken Chen, Jinsong Wang, etc, A New Method of Gait Generation for A Biped Walking Robot[C]. Proceedings of the IEEE_RAS International Conference on Humanoid Robots, pp.295-302, 2001.

[2] Philips Semiconductor, SJA1000 Stand-alone CAN Control Application Note [Z], pp.3, 1997.
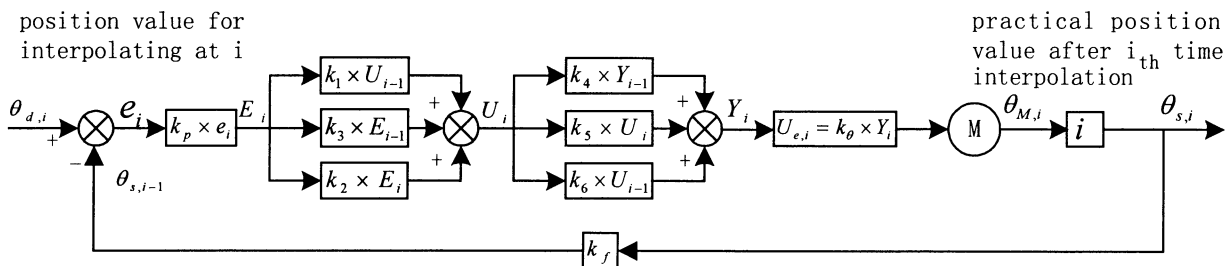
Fig. 7 The control algorithm model of position control card

The Development of a Biped Humanoid Robot----THBIP- I

ZHAO Jian-dong[1]

1. Department of
Precision Instruments and Mechanology
Tsinghua University
Beijing, China. 100084
zhaojdong@mails.tsinghua.edu.cn

WANG Ji-wu[1],
Zhang Wei[2],
SHAO Li-jun[1],
CHEN Ken[1]
2. DigitalChina Co.
Beijing, China. 100083

Abstract

In this paper we present the configuration, control system, gait planning, and experiment of the biped humanoid robot─THBIP- I , which possesses of anthropopathic external form and skeleton. It has 32 DOFs in all, and the mechanical gearings of some key joints are based on screw-nut system. As to control system, THBIP- I adopts distributed hierarchical architectures, consisting of organization layer, coordination layer and execution layer. And the CAN bus interface is employed between the two latter. Meanwhile, the robot walking gait is planned according to the expanded ZMP point on virtual plane, supposed between the left foot plane and the right one, so the ZMP track of actual robot motion conforms to the desired one. In conclusion, the walking experiment photos strongly demonstrate the whole research work, and the final goal is to improve the dynamic walking speed, stability and balance, and the intelligent degree of this robot.

Keywords

CAN Bus, Expanded ZMP, Coordination Controller

1 Introduction

Humanoid robot is one of the most vibrantest research directions in robot field. It has similar function of human being, and integrates with apperceive, decision-making, activity and alternation. With a view of development of science and technology, and giving service to human being, we began to research a biped humanoid robot── THBIP- I from May 2000.

At present, THBIP- I is the most excellent biped humanoid robot in design level and anthropopathic motion in China. Fig.1 shows the overview of prototype configuration. It is 174mm in height, 122kg in weight, and has 32 degree of freedoms, as Fig.2 shows, which are 2 of head, 6x2 of arms, 3x2 of hands.

Considering of bionics [1], minimum joint torque and essential walking function, 3x2 DOFs (pitch, yaw and roll) are settled for hips, 2x2 (pitch and roll) for ankles, and 1x2 (pitch) for knees. Derived from the result of HONDA biped robot experiment, when walking, it is the heel and toe for supporting body. So there has no toe joint for prototype. Thus robot may stand on uneven surface, change the walk direction, span step and satisfy the sudden discontinuousness of walking. So it can realize flat walking, slope walking, spanning step, single feet support, single feet swing and squat, etc. Because ankle has no roll joint, it needs 4-5 steps to rotate 90 degrees.

Two CCD video cameras and a stereo microphone are settled inside head, for environment scanning and voice recognizing separately. A notebook computer, coordination controller, is located into the trunk; an on-board battery system is set on the back; a gyroscope lies in the center of two coxae, for gaining the center of gravity of robot; and two Universal Force-Moment
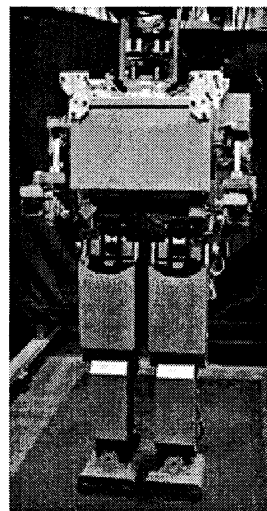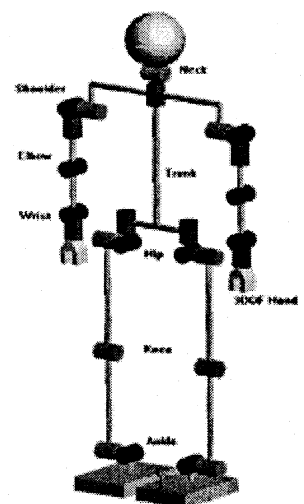
Fig. 1 THBIP- I   **Fig. 2 Configuration of THBIP- I**

Sensors are built in two ankles, for measuring the ground reaction force. For each joint, a potentiometer is used for obtaining the present absolute position.

In mechanical design, upper limb uses worm-wheel transmission. For low limb, the screw-nuts gearing is different from the robot of HONDA [2] and SONY [3]. It can reduce transmission radio and friction damp, and provide high torque and high velocity.

2　Control System

From the point of view of the intelligent control, and based on fieldbus technology, THBIP- I adopts hierarchical structure and distributed control system, consisting of organization layer, coordination layer and execution layer, and the configuration and function of each layer is shown as Fig.3.

The first layer is located outside of prototype, and takes charge of system management, human-machine interface, project assignment and off-line simulation.

The second one, embedded into the trunk, consists of a notebook computer and two PC-104 computers. Two CCD video cameras connect with notebook computer through two USB ports. Also the voice recognition system connects with notebook computer through COM1. The upper limb controller (PC-104 computer) controls the motion of two arms; the low limb controller
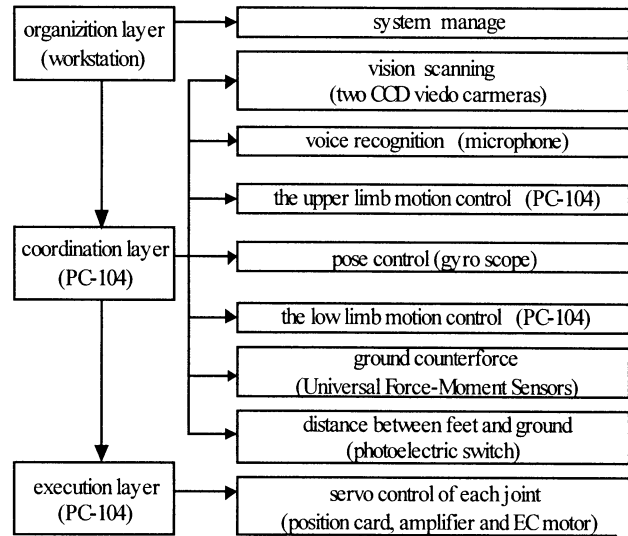
Fig. 3 Structure of control

(PC-104 computer) controls the motion of two legs. Combined with a 16-bit AD convert card, the pose computer (PC-104) gains the data of gyroscope, and calculates the center of gravity of THBIP- I, which is sent to the low limb controller via ISA. These data are used for ZMP planning and on-line gait modification. All of the coordination computers compose of a LAN through Ethernet card, and coordinate the motion control between arms and legs. The organization layer and coordination one are operated via wireless communication. Between coordination and execution,
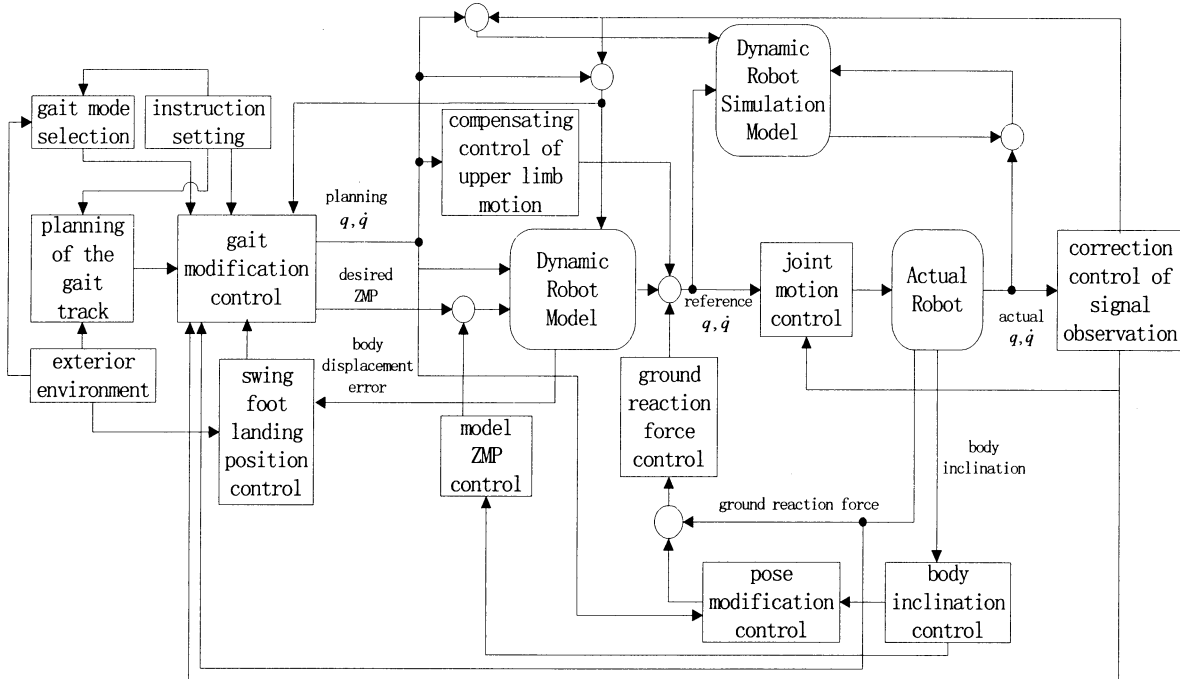
Fig. 4 The software structure of harmony control system

the authors make a success of applying CAN bus interface to humanoid robot for the first time.

The software structure of coordination control system is shown as Fig.4. Robot has three modules: dynamics calculation module, dynamics simulation module, and real robot module. The main control modules include gait planning, gait modification, joints motion control, and model ZMP control, etc.

The execution layer completes the joints servo control of the upper limb and the low limb. In view of many joints message, environmental interference and field equipments, Based on CAN bus, the first author designs the distributed servo control system. The joint servo unit is composed of position control card, numeral servo amplifier with speed loop and current loop, and EC motor. All of them are settled into the robot. The position control card includes a 16-bit MCU, D/A, A/D, I/O port, and CAN bus interface. On the basis of communication protocol, the execution layer receives instructions from the coordination layer. Each node analyses the instructions, and runs corresponding follow. The other fieldbus nodes also collect message, communicate with master or other nodes, and share data with them.

3 Gait Planning and the Expanded ZMP

Because the actual planes contacting with robot sole are two or more planes, of which the normal heights and directions are different. The researchers supposed that there is a virtual plane varied between the first plane (left foot) and the second (right foot), and the desired ZMP is supposed on virtual plane. Fig.5 shows the foundation of virtual coordinate system and the definition of expanded ZMP point, which equals to the E_{31} coordinate origin. The x-y plane of the first coordinate system is the first foot sole plane, the coordinate origin is located at the junction of ankle joints; the x-axis direction is same to robot forward direction; the
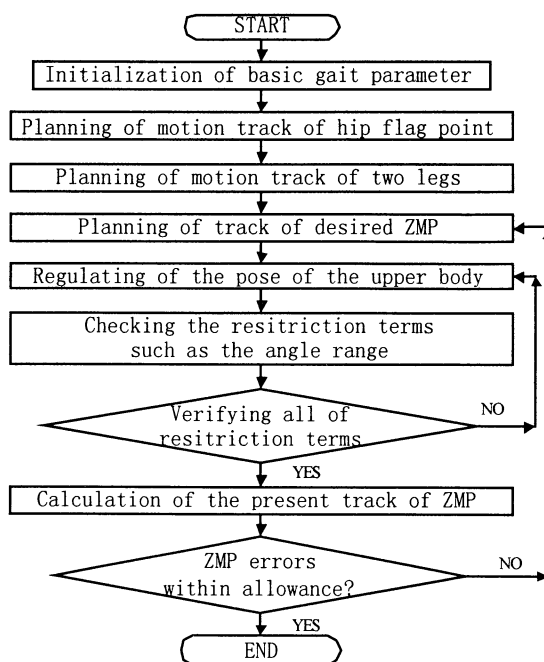
Fig. 6 Planning of Gait Based on Expanded ZMP

z-axis is the normal direction of ground and points to the outward of ground. The foundation of Virtual plane commendably solves the transition from flat plane to incline plane, and the problem of going to or downing stairs. Also in actual flat walking, the error produced by uneven surface can be eliminated. (Conferring reference [4] for details).

In addition, there has no uniform criterion in body modification, so on condition that robot walking speed is not high, the position of center of gravity of the whole system is selected as reference to instructing to modify the angle data of body. At
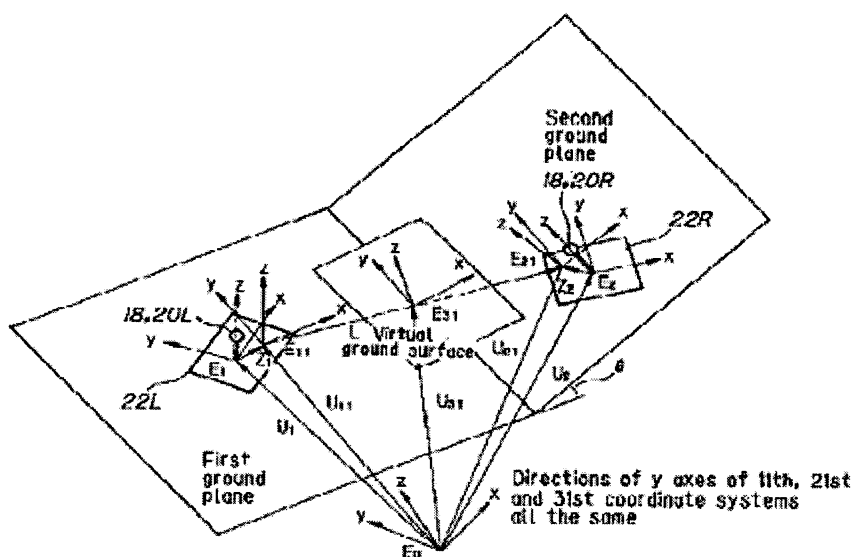
Fig. 5 Definition of Expanded ZMP

- 22 -

the same time, the track of ZMP and the center of gravity are smoothed as possible, and the latter inclines to corresponding point of static state.

Then as Fig.6 shown, gait is planned according to the desired ZMP point, meanwhile, make the ZMP track of actual robot motion conform to the desired one. Thus, the new gait plan system not only can make robot walks on uneven surface steadily, but also can maintain the continuity of composition of ground reaction forces and moments, and ensure the smoothness and continuity of robot motion.

Now, THBIP- I can independently and continuously walk 4 step by the speed of 8seconds/step, and the stride is 200mm. As Fig.7 shown, the photos demonstrates the process of static gait walking experiment in detail, including two legs kneebend, moving center of gravity, single leg support, single leg swing; two legs support; and two legs stand.

4 Future Plan

In the middle of April, 2002, without manpower support, THBIP- I can execute static walking independently and continuously, one gait is 8 steps by the speed of 20s/step to 12s/step, the stride is 200mm. The other gait is 4 steps by the speed of 8s/step, and the stride is 300mm. More and more experiments have proved that the whole work is successful, and the distributed control system based on CAN bus is reliable, and the screw-nuts gearing mechanism is favorable.

In the research of biped humanoid robot, the critical technologies of anthropopathic walking are the motion capacity of joints and the balance of walking. The final goal of THBIP- I will realize flat walking, uneven surface walking,

spanning step and swerving. And the stride is 200-300mm, the static walk is 10-20s/step, the quasi-dynamic walking is 10-20s/step, and the dynamic walking is 0.5-2s/step. In addition, it is integrating sensors system for improving the intelligent degree of the robot.

Reference

[1] Fred R. Sias, Jr. and Yuan F. Zheng, How Many Degree-of-Freedom does a Biped Need? [C]. IEEE International Workshop on Intelligent Robots and Systems, IROS '90, pp. 297-302, 1990.

[2] Kazuo Hirai, Masato Hirose, Yuji Haikawa, etc. The Development of Honda Humanoid Robot[C]. Proceedings of the 1998 IEEE International Conference on Robotics & Automation Leuven. Belgium, pp.1321-1326, 1998.

[3] Yoshihiro Kuroki, Tatsuzo Ishida, Jin ichi Yamaguchi, et al. A Small Biped Entertainment Robot[C]. Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pp.181-186, 2001.

[4] Zhang Wei. Research and Simulation on Gait of Humanoid Biped Walking Robot, master thesis, Tsinghua University, China, 2002, 6.
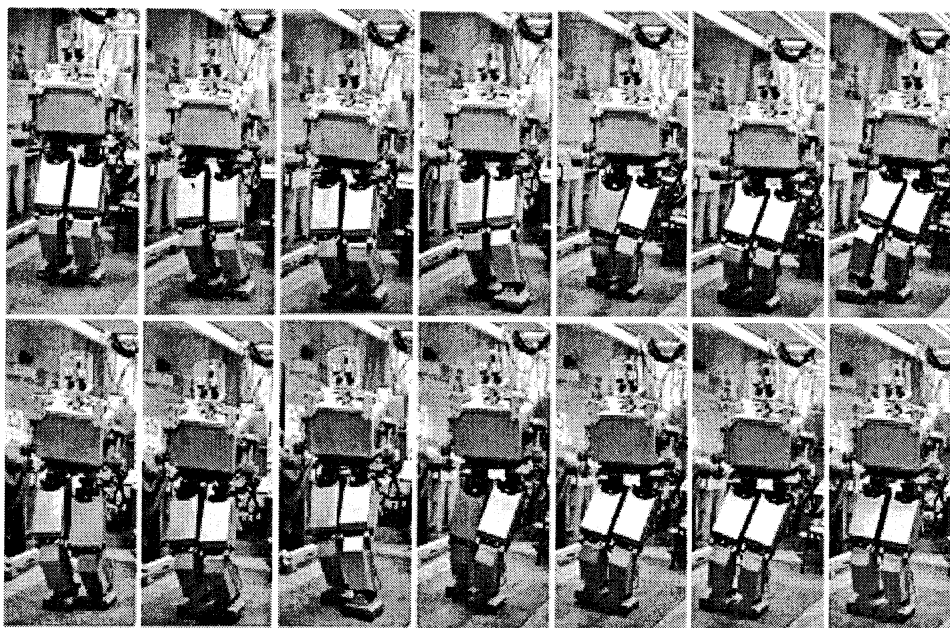
Fig.7 gait experiment

Deadlock Avoidance Method for Multiagent Robot System Using Intermittency Chaos

Yoichiro Maeda

Department of Human and
Artificial Intelligent Systems,
Fukui University,
Bunkyo, Fukui 910-8507

Takayasu Matsuura

System Administration Section,
Technical Services Department,
Japan Network Information Center
Chiyoda-ku, Tokyo 101-0047

Masaharu Mizumoto

Department of Engineering Informatics,
Faculty of Information Science and Technology,
Osaka Electro-Communication University,
Neyagawa, Osaka 572-8530

Abstract

In this research, we regard deadlock phenomenon in multi-agent robot toward the destination while avoiding plural obstacles as a local minimum solution in a search problem. We propose the method that can avoid deadlocks efficiently, by changing singleton values of fuzzy rule that described the action in an individual agent in accordance with the intermittency chaos with the modified logistic map. Furthermore in this paper, we report the result of confirming the efficiency of our proposed method by using the multi-agent robot simulator.

Keywords: Intermittency Chaos, Deadlock Avoidance, Logistic Map, Multiagent Robot, Fuzzy Rule

1 Introduction

Recently, the chaotic theory is frequently used in the engineering field of information processing and robotics. Y.Nakamura et al. [1] confirmed that an autonomous mobile robot can move chaotically by applying Arnold equation to the motion equation of a two-wheeled mobile robot. Matsumura [4] proposed an obstacle avoidance method that makes a few loss in locomotive distances and consumption energies by the intermittency chaos.

Recently, the chaotic theory is frequently used in the engineering field of information processing and robotics. Y.Nakamura [1] confirmed that an autonomous mobile robot can move chaotically by applying Arnold equation to the motion equation of a two-wheeled mobile robot. K.Matsumura [2] simulated the obstacle avoidance motion for an autonomous mobile robot using the intermittency chaos with the modified Bernoulli system.

In this research, we regard deadlock phenomenon in multi-agent robot toward the destination while avoiding plural obstacles as a local minimum solution in a search problem and attempt to avoid this problem efficiently by the chaotic method. We have already proposed the method that can avoid deadlock as a group efficiently, by changing singleton values of fuzzy rule that described the action in an individual agent in accordance with a network of chaotic elements.

Network of chaotic elements is a lattice model connected many elements that have chaotic behavior. The most typical models are Globally Coupled Map (GCM) structure that every element is connected globally and

Coupled Map Lattice (CML) structure that two different neighboring elements are connected locally proposed by K.Kaneko [3].

We produced a multi-agent robot simulator in the research and reported the result of confirming the efficiency of our proposed algorithm by using our simulator [4] [5]. As a result, we confirmed that each agent gradually becomes possible to avoid deadlocks, but in the method using GCM we found that is difficult to take efficient avoidance action because a change of one agent have an influence on all agents. However, in CML, we confirmed that each agent was possible to avoid efficiently as a whole group.

In this paper, we propose the method that can avoid deadlocks efficiently, by changing singleton values of fuzzy rule that described the action in an individual agent in accordance with the intermittency chaos with the modified logistic map. Furthermore we performed the simulation to confirm the efficiency of our proposed method by comparing with the modified Bernoulli system. In this paper, we report the result of confirming the efficiency of our proposed method by using the multi-agent robot simulator.

2 Fuzzy Obstacle Avoidance Model

At first, we establish a multi-agent robot model toward the destination while avoiding the obstacles and the other robots. The robot carries out sensing of the circumference and computes the relative distance to obstacles within the sensor area and the relative angle to the obstacles in the viewpoint of a robot with its coordinates.

Each obstacle avoidance angle is calculated by the fuzzy reasoning for every obstacle and the avoidance vector is decided by these angles. The robot calculates the destination vector toward a desired point and computes a composite vector weighted according to the distances to the obstacles. This composite vector is equal to steering vector for the robot.

The vectors to avoid all obstacles within the sensor area are calculated by the simplified fuzzy rules. Robot moves to the direction for a unit vetor which is compounded with some avoidance vectors weighted according to the distance between a robot and an obstacle. Membership functions of three labels for the distance and six labels for the direction, and singletons of six labels for the steering are described in the fuzzy rule.

3 Modified Logistic Map

In this research, we propose the intermittency chaos with the modified logistic map for changing singleton values of fuzzy rule to avoid deadlocks efficiently. This function has a symmetrical shape for the ordinary logistic map, but has also same structure of the logistic map.

$$x(t+1) = \begin{cases} ax(t)(1-x(t)) & (0.0 \le x(t) < 0.5) \\ -ax(t)(1-x(t)) + 1 & (0.5 \le x(t) \le 1.0) \end{cases} \quad (1)$$

a : Parameter to decide the degree of chaotic behavior
   $(1.0 \le a \le 4.0)$
$x(t)$: Output value of modified logistic map at time t

In the equation (1), $x(t)$ shows a value of the function at time t and a shows a parameter which decides the chaotic behavior between $1.0 \le a \le 4.0$. As same as the ordinary logistic map, the values of this function gradually became chaotic when a parameter a approaches 4.0. Figure 1 a) shows a return map with the horizontal axis $x(t)$ and the vertical axis $x(t+1)$. We confirmed that the modified logistic map function has a value of the parameter a which generates the intermittency chaos similar to the modified Bernoulli system.

The attractor of this modified logistic map is shown in Figure 1 b). As shown in this figure, the system shows chaotic behavior when the value of parameter a is between 3.57 and 4.0. Specially, this attractor is symmetrical vertically for the horizontal line $x(t+1) - x(t) = 0$.

a) Return Map   b) Attractor
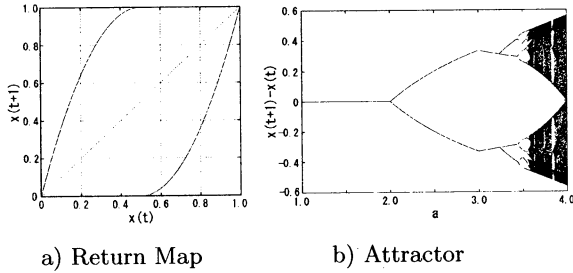
Figure 1: Modified Logistic Map

Furthermore, blank windows in a chaotic part of the attractor between $3.57 \le a \le 4.0$ are called edge of chaos. In this blank windows, the modified logistic map has the characteristics of the intermittency chaos which intermittently repeats a periodical behavior and a chaotic behavior.

Figure 2 a), b), c) show the typical trajectory of the modified logistic map in time scale with a=3.63557, 3.7453, 3.8575, respectively. In the figure, the part of a periodical behavior is called 'laminar' state and the part of a chaotic behavior 'burst' state. Each trajectory has the characteristics as shown in Table 1.

Table 1: Characteristics of Intermittency Chaos

Characteristics	$a = 3.6356$	$a = 3.7453$	$a = 3.8575$
Laminar State	short	short	long
Burst State	short	long	midium
Periodicity	8	8	6
Intermittency Interval	short	long	midium

a) a=3.63557   b) a=3.7453   c) a=3.8575
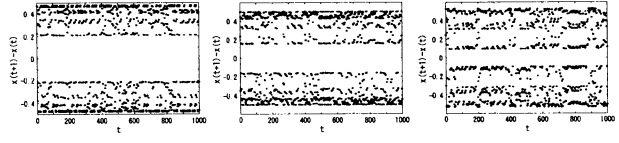
Figure 2: Trajectory of Modified Logistic Map

4 Fuzzy Obstacle Avoidance Method by using Intermittency Chaos

Deadlock avoidance behavior of the agent robot is realized by giving a chaotic vibration of the above-mentioned modified logistic map to the singleton in THEN part of fuzzy rule. Chaotic vibration is given by the following equation (2).

$$F_{label}(t+1) = F_{label}(t) + \alpha(x(t) - x(t-1)) \quad (2)$$

$F_{label}(t)$: Singleton value of obstacle avoidance angle
α : Chaotic vibration gain

In this research, three typical types of the intermittency chaos mentioned in the last section are given to the singleton of fuzzy rule according to the condition of the deadlock. Intermittency chaos proposed in this paper repeats a laminar or a burst part, but laminar part performs normal obstacle avoidance behavior and burst part shares random search behavior. We define the judgement mechanism of the deadlock condition according to the number of nearest obstacles O_n.

In case of $O_n = 1$, that is, when a robot came across only one obstalce, $a = 3.7453$ with long burst states is selected. In this case, an agent robot can avoid the deadlock by using choatic behavior with random vibration.

Next in case of $O_n = 2$, there are two conditions, that is, when a robot can not go forward physically and when a robot stops because the direction of the steering angle obtained by fuzzy obstacle avoidance rule is equal to the direction an obstacle exists. In the first case, that is, when the interval of two obstacles O_{wide} is smaller than an certain distance W_D, we use the intermittency chaos with $a = 3.63557$ which the chaotic parameter a continuously repeats burst and laminar states. By using this method, a deadlock escape motion by the random vibration of the burst state and an obstacle avoidance behavior by the stable vibration of the laminar state are repeated at short interval.

In case of O_{wide} is larger than an certain distance W_D, $a = 3.8575$ is used as a chaotic parameter. In this case, the time width of burst state for the deadlock avoidance is shorter than the case when O_{wide} is small.

5 Judgement of Deadlock Condition

In this research, whether an agent robot falls into a deadlock or not is judged by the following algorithm. Deadlock condition is recognaized by the vector for an absolute position $\vec{P_r}(t)$ of an agent robot.

Step 1: If $|\sum_{i=t-m}^{t} \vec{P_r}(i)|^2 < \epsilon$ and $t > m$, then robot recognizes to stop at same position and T_s is increased only m.

Step 2: If T_s is larger than T_d, then go to Step 3 else go to Step 1.

Step 3: Deadlock avoidance mode is started and chaotic vibration is given to fuzzy obstacle avoidance rule. And T_a is started to count.

Step 4: If $|\sum_{i=t-T_a}^{t} \vec{D_L}(i)|^2 > w$, then deadlock avoidance is successful and go to Step 1 else go to Step 5.

Step 5: If T_a is larger than T_{deep}, then robot recognizes to fall into a deep deadlock, α is increased only $\Delta\alpha$, w is increased only Δw and go to Step 4.

T_s : Total time robot is stoping at a same position
m : Positive constant value
T_d : Time constant to judge falling into a deadlock
$\vec{D_L}(t)$: Absolute vector when robot falls into a deadlock
T_a : Total time robot is falling into a deadlock
T_{deep} : Time constant to judge falling into a deep deadlock
ϵ : Constant value to judge stoping at a same position
w : Constant value to judge depth of deadlock

By this algorithm, deadlock condition is recognized by using only local informations an agent robot can obtain. Furthermore, we can change the chaotic vibration according to the depth of deadlock by the parameter T_a.

6 Simulation

We performed simulation to confirm the efficiency of our proposed method by comparing the following intermittency chaos with the modified Bernoulli system [2].

$$x(t+1) = \begin{cases} x(t) + 2^{B-1}(1-2e)x(t)^B + e & (0 \le x(t) \le 0.5) \\ x(t) - 2^{B-1}(1-2e)(1-x(t))^B - e & (0.5 < x(t) \le 1.0) \end{cases}$$
(3)

B : Parameter of chaotic degree
e : Positive small constant value

Trajectories of the modified Bernoulli system when the parameter $B = 1.0, 2.0$ in the equation (3) are shown in Figure 3. As shown in this figure, the chaotic degree becomes high and burst states occupy the majority when the parameter B approachs to 1.0. On the other hand, the system becomes gradually intermittent when the parameter B approachs to 2.0.

a) $B = 1.0$ b) $B = 2.0$
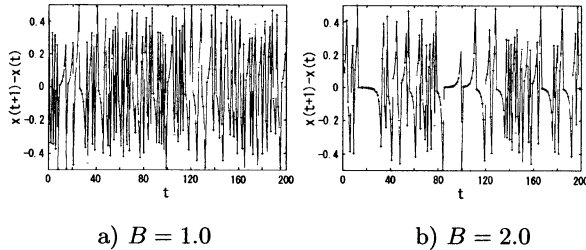
Figure 3: Trajectories of Modified Bernoulli System

In this simulation, we applied the simulated annealing method that deadlock avoidance is performed by using the trajectory in $B = 1.0$ and B is gradually approached to 2.0 according to the total time an agent robot is falling into a deadlock.

We performed simulation by using the following three method. Simulation parameters are also shown below.

- CASE1: Modified logistic map
- CASE2: Modified Bernoulli system
- CASE3: Random search

Velocity of Agent v	constant
Initial Agent Position	random
Initial Obstacle Position	random
Number of Agents A_n	20
Distance of Sensor Limitation S_r	30.0
Range of Sensor	180 degrees forward
Priority Area for Obstacle Avoidance δ	12.0
Expiration of Simulation	$Time \ge 2000(sec)$

Table 2 shows the results of simulation in each case. Each number in the figure shows the average time for ten trials of simulation that an agent reached at a goal area within a constant time (Time $\le$ 2000) was falling into deadlock. Some area without number shows there is the agent which could not reach at goal area.

Table 2: Simulation Results

Type of Vibration	Number of Obstalce					
	100	120	140	160	180	200
CASE1	136.1	226.3	228.9	257.4	288.2	409.3
CASE2	221.2	296.1	296.8	343.5	390.5	-
CASE3	249.5	654.5	671.4	-	-	-

Each total time of falling into deadlock in case of 120 and 140 obstacles is shown in Figure 4, 5. Figure 6 shows transitions of the distance between each agent robot and a goal aerea, and Figure 7 shows real trajectories of agent robots in the simulation field in case of 140 obstacles.

In case of the simulation using the modified Bernoulli system, we confirmed that total escape time of avoiding obstacles is shorter than that of random search and all agent reach at a goal area up to 180 obstacles as shown in Figure 4, 5. The intermittency chaos generated by the modified Bernoulli system is more efficient in deadlock avoidance than the random search. However, it is difficult to decide the parameters about the annealing time according to the condition of simulation.

In case of the simulation using the modified logistic map, we confirmed that this method is more efficient in the deadlock avoidance behavior than the other two methods. By using chaotic vibration of the proposed modified logistic map, the agent robot can perform the symmetrical search of the avoidance angle for the moving direction. Furthermore, we found the modified logistic map has better performance than the modified Bernoulli system because of the periodicity of the laminar part in the proposed method.

7 Conclusions

In this research, we regard deadlock phenomenon in multi-agent robot toward the destination while avoiding plural obstacles as a local minimum soluti on in a search problem and attempt to avoid this problem efficiently by the chaotic method. We proposed the method that the multi-agent robot can avoid deadlock as a group efficiently by giving the chaotic vibration of the modified logistic map to singleton values of fuzzy rule that described the action in an individual agent. And we performed a multi-agent robot simulation to confirm the efficiency of our proposed method. By the result of simulations in case of the proposed method, the modified Bernoulli system and random search, we confirmed that the agent robot with chaotic vibration of the proposed method performs the deadlock avoidance behavior most efficiently.

As a future work, we consider to perform the simulation for the other complex problem and to apply the proposed method to the experiment by real robots.

References

[1] Y.Nakamura and A.Sekiguchi, "The Chaotic Mobile Robot," Journal of Robotics Society of Japan, 15(6), 918-926, 1997. [in Japanese]

[2] K.Matsumura, "Simulation of Moving Obstacles Avoidance by the Orbit Control Method Using Intermittency Chaos," Transactions of the Institute of Electronics, Information and Communication Engineers, Vol.J81-A, No.5, pp.870-880, 1998. [in Japanese]

[3] K.Kaneko, "Chaos, CML, Complex System," Science, 62(7), 427-435, 1992. [in Japanese]

[4] T.Matsuura and Y.Maeda, "Deadlock Avoidance of a Multi-Agent Robot Based on a Network of Chaotic Elements," Advanced Robotics, 13(3), 249-251, 1999.

[5] Y.Maeda, "Deadlock Avoidance Method for Multi-Agent Robot System Using Network of Chaotic Elements," Proc. of IEEE International Symposium on Computational Intelligence in Robotics and Automation, pp.282-283, 2001.
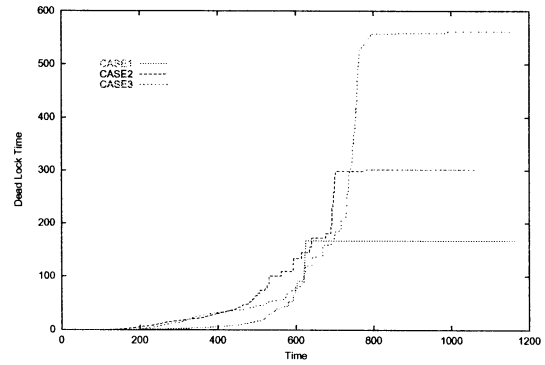
Figure 4: Deadlock Staying Time (120 Obstacles)
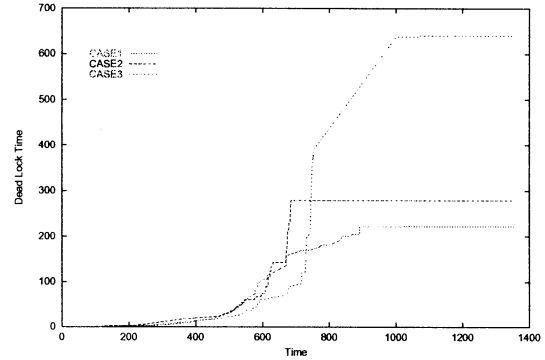
Figure 5: Deadlock Staying Time (140 Obstacles)
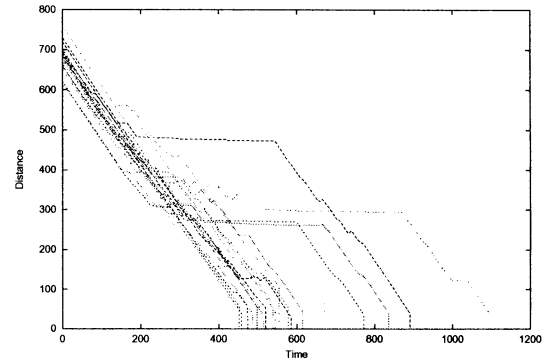
Figure 6: Distance Transitions between Agent and Goal (140 Obstacles)
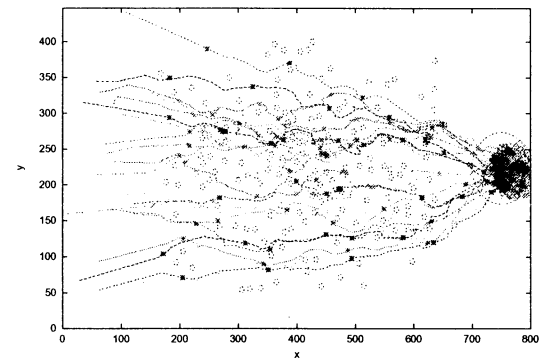
Figure 7: Trajectories of Agent (140 Obstacles)

The internal model of the other for learning the cooperative behavior

Keisuke Kondo
Dept. of Economics
Kyoto Gakuen University
Kameoka, Kyoto 621-8555 Japan

Ikuko Nishikawa
Dept. of Computer Science
Ritsumeikan University
Kusatsu, Shiga 525-8577 Japan

Abstract

The goal of this research is to design an intelligent agent which learns to behave cooperatively with other intelligent agents to carry out a common task. First, the present paper shows an experimental result. In the experiment, a human subject is required to perform a task on a computer terminal. The experiment is done under the following 3 conditions given to the subject. A subject executes 12 trials under each of 3 conditions, and the performance is measured by the time required to attain the task, namely, to move the object to the goal. The result strongly suggests that the modeling of another agent is essential for a successful cooperative behavior, and that more information on the agent through its direct detection is not necessarily helpful to develop the model. Next, a learning agent is designed as a layered neural network, which outputs the actions of holding and moving from inputs of sensory data on the carrying object and the surrounding environment with several obstacles. Based on the above experimental observation, a module to predict the motion of another agent is added to the network, which has a potential to learn an internal model of the other agent. The neural network learns via an unsupervised method through the repetitive processes of the same task as the above experiment of a human subject.

1 introduction

According to the study by Baron-Cohen etc.[1], the hypothesis that the autism is owing to the "theory of mind" is proposed. In 1978, David Premack and Guy Woodruff[2] who paid attention to the fact that the primates conduct the behavior as if they surmised their company's condition of mind, and called this kind of behavior the "theory of mind." In order to enable social behavior, the system to forecast the behavior of the partner is said to be necessary, however even in the case to conduct a simple cooperative behavior, it is thought to be effective to have the system to forecast the thought(behavior) of the partner. Therefore first this study verifies whether a man needs to have the system to forecast the partner's behavior, resorting to the simulation that the two robots conduct the cooperative behavior of carriage of an object. Next we show the validity for cooperative behavior, by mounting the system in the robots to forecast partner's behavior.

2 The human cooperative behavior

In order to verify whether even in the cases to conduct a simple cooperative behavior, below-mentioned simulations are experimented.

2.1 Purpose

The purpose is to carry an object to the destination by the cooperation of two robots. One robot is controlled by a computer and the other is controlled by a man to measure the time for reaching the destination.

2.2 Environments

The robots move by the wheels without the dynamic influence from the floor and an object. Owing to the positive and negative values of two motors on both sides of right and left as well as the size relation between the values, the robots can move in 8 ways, namely forward, forward to left, forward to right, round to left, round to right, back to left, back to right and back, as shown in the Figure 1 to make a choice of behavior from 8 options by the mouse.
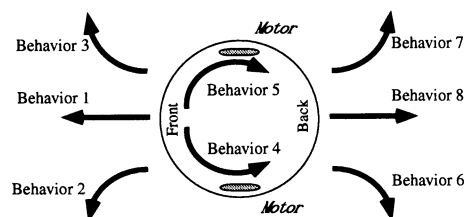
Figure 1: Eight type of behavior

In this study, two simulations are proposed, namely one for exercising and the other for conducting cooperative behavior. The simulation image for exercising is displayed in the Figure 2.
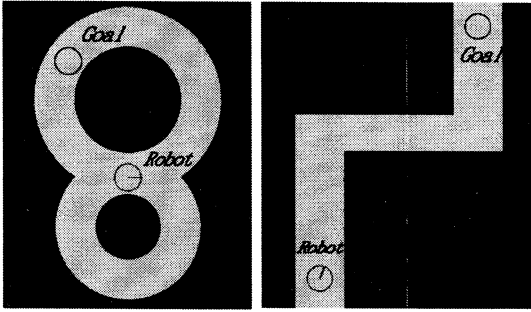
Figure 2: Simulation of control practice

This figure shows that the singluar robot moves to the goal and evade the obstacle to be trained for the training to control the robot.

The simulation image for conducting cooperative behavior is displayed in the Figure 3. As for the object, the robots hold both edges and in case where the robots get farther or nearer more than in a measure, they are supposed to drop the object. The subject moves to the goal, controlling the part of an object of the different color to overlap the indication of the robot.
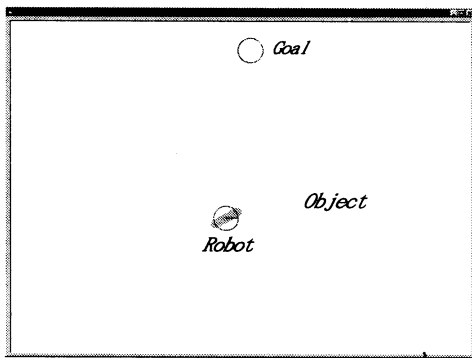
Figure 3: Experiment of cooperative behavior

The robot under the control by the computer has the same function, restricted to conducting the simple following behavior. The simple imitation behavior is realized by the following 3 rules.

- In case where the object is attracted to the right side against the advancing direction or pushed to the left side, it moves forward to right.

- In case where the object is attracted to the left side against the advancing direction or pushed to the right side, it moves forward to left.

- Other than two cases mentioned above, the object moves forward.

On account of the movement that is composed of 3 behaviors, namely forward, forward to right and forward to left, without paying sufficient attention to controlling the other robot, the object will be dropped.

2.3 Method of experiment

In the first place, the subject conducts moving the robot to the destination and evading the obstacle to get accustomed to the operation of his/her robot. The 12 sorts of environments of diverse initial conditions and the location of goals are provided and the time of reaching the destination in the respective environment under the following 3 conditions is measured.

(a) Without displaying the counterpart robot nor indicating its existence.

(b) Without displaying the counterpart robot and indicating only its existence.

(c) Displaying the counterpart robot.

In the first step of the course of experiment, the subject gets accustomed to the control of the robot sufficiently, which is judged by the self report of the subject. Next the purpose is indicated on a panel, and its content is as follows: "Please move to the destination without dropping the object. If the robot parts from the part of the object of the different color, it is equivalent to dropping the object. The object jitters freely, so please pay attention not to drop it. The time to reach the destination is measured." After measuring the time in the 12 sorts of environments under the condition (a), the next purpose is indicated on a panel, and its content is as follows: "To be frank with you, on the opposite side of the object there exists a robot similar to yours, moving so as not to drop the object. On the display the robot is not indicated, however please move to the destination considering it. The time to reach the destination is measured."

After conducting the same simulation under the condition (b) again, the next purpose is indicated on a panel, its content is as follows: " This is the last measurement. This time the counterpart robot is indicated on the display. Please move to the destination again. The time to reach the destination is measured." The simulation with the display of the counterpart robot is held to measure the time under the condition (c) again.

2.4 Result

The result of experiment conducted by 4 subjects is shown in the Figure 4.
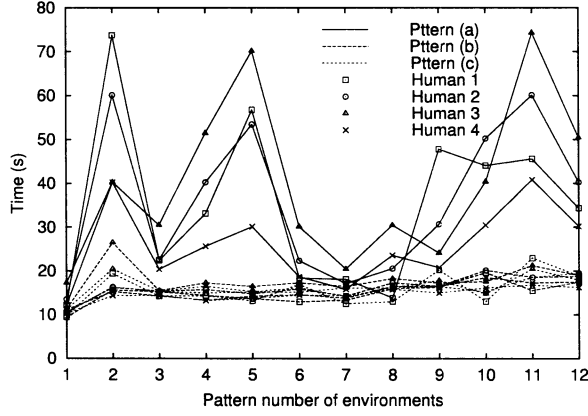
Figure 4: The result of expriment by 4 subjects

Because the behavior of the counterpart robot is diverse according to the difference between the initial location and the location of goal, there are facile cases and difficult cases. However, among the cases of (a),(b) and (c), the time to reach the destination apparently differentiates, the 4.7 times of while is necessary at most. It is difficult to forecast the next behavior under the condition (a) in the observation of the movement of an object, but it is considered that the time to reach the destination is consequently shortened under the condition (b), where the existence of the counterpart robot is recognized to forecast the movement of an object with ease. From this result, it is considered that the establishment of the internal model to recognize the existence of the partner and to forecast the movement of the partner is an effective method, despite the cooperative behavior such as the simple carriage of an object.

3 Cooperative behavior by robots

3.1 Simulation model

To establish in a robot the internal model to forecast the partner to build up the learning system of cooperative behaviors. The simulation is conducted in the same environments as the Chapter 2, namely the robot under the control of the computer conducts the imitation behavior under the same condition as the Chapter 2. To build up the learning system of cooperative behaviors for the robot under the control of

a man to let it learn. As is shown in the Figure 5, the information the robot can obtain is the direction of the goal and that of an object both of them against the advancing direction as well as the distance of movement of an object. No direct communication between robots is set up.
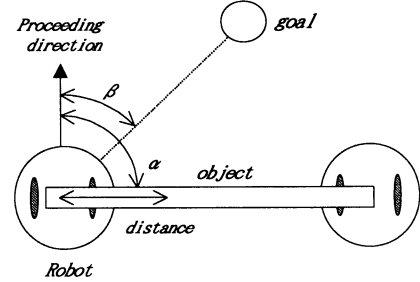
Figure 5: The robot and object

3.2 Learning network

The network to select a behavior out of the 8 options according to the condition of an object and the information of the goal is shown in the Figure 6.

The upper side of the Figure 6 is the network composed of 8 input units $x_i(i = 0 \ldots 7)$ based on the object and goal as well as 8 output units $y_j(j = 1 \ldots 8)$ corresponding to the respective behavior. The input value into x_i is the normalization of 0 to 100. The lower network of Figure 6 is the internal model to forecast the behavior of the counterpart robot.

Hereby the input-output relations are expressed in the form of a weight-sum with the following total combination.

$$y_j(t) = \sum_{i=0}^{7} w_{ij} x_i(t) + \sum_{k=0}^{7} w_{kj}(t) z_k(t) \qquad (1)$$

At this time the behavior J corresponding to the largest y_j is chosen.

The payment is given according to the Hebb type learning-behavior results on the basis of the payment, the weight values w_{ij} and w_{kj} are updated by the Hebb type learning.[3][4] The payment obtained at the time t is distributed in a discount toward the behavior at the same time and a past series of behaviors. The learning by the payment $r(t)$ is given in the following formula.

$$w_{ij}(t+1) = w_{ij}(t) + \alpha r(t) \sum_{n=0}^{100} \gamma^n x_i(t-n) \delta_{jJ(t-n)} \quad (2)$$

The internal model of the other
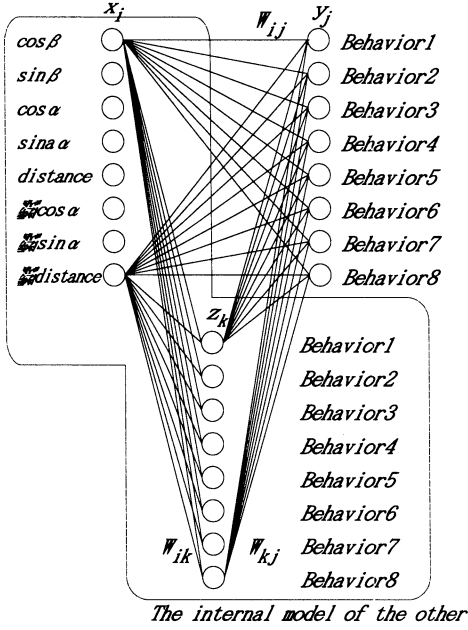
Figure 6: Learning network

$$w_{kj}(t+1) = w_{kj}(t) + \alpha r(t) \sum_{n=0}^{100} \gamma^n z_k(t-n)\delta_{jJ(t-n)} \quad (3)$$

α is the learning rate, $\gamma(<1)$ is the discount rate, and δ_{ij} is the Kronecker's delta. The payment $r(t)$ is determined by the arrival to the destination(>0), the approach to the destination(>0), the risk of dropping the object(<0), the dropping of an object(<0). The part of internal model is used to analyze the movement of the counterpart robot from the later results and is used as the teacher data(z_{teach})and is made to learn in the following formula.

$$w_{ik}(t+1) = w_{ik}(t) + \beta(z_{teach} - z_k)x_i \quad (4)$$

3.3 Cooperative learning's result

The 12 sorts of different initial locations and destinations are bundled as the singular trial and the robots are made to learn by 20 trials. Its result is shown in the Figure 7. After the 20 trials the average reaching time comes to be 16 seconds, which is almost the same result as in the cases of humans. Additionally, the accuracy rate expressing whether the internal model exactly functions or not is making progress by degrees according as the reaching time gets shorter.
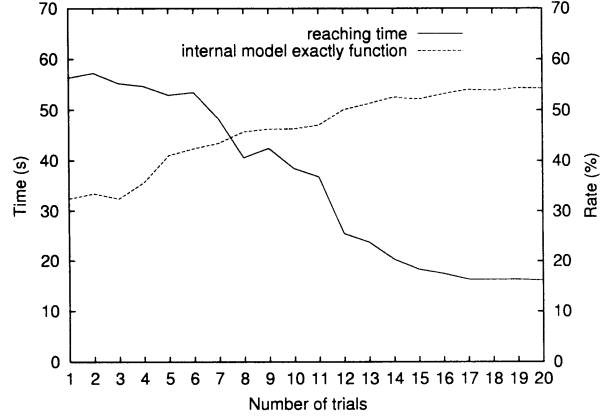
Figure 7: The result of cooperative behavior experiment

4 Conclusion

Using simulations, this study verifies that it is necessary for a human conducting cooperative behavior to recognize the partner and build up the internal model even in the case of the simple carriage of an object. By establishing the internal model to forecast the behavior of the counterpart robot in the cooperative behavior learning by 2 robots, it comes to be possible to acquire the cooperative behavior. It is expected to be effective for the cooperative and social behavior among plural numbers of robots to build in the system to forecast the partner's behavior based on the "theory of mind."

References

[1] Simon Baron-Cohen,*Mindblindness:An Essay on Autisum and Theory of mind*, The MIT press, 1995

[2] Masuo Koyasu, "Iwanami Libraryof science: Theory of mind", Iwanami press, 2000.

[3] K. Kondo, I. Nishikawa and H. Tokumaru, " A study on an autonomous mobile robot control with behavior network", *Int. Symp. on Artificial Life and Robotics*, pp. 80-83, 1997.

[4] K. Kondo, I. Nishikawa, " Hebbian learning of behavior network and its application to a collision avoidance by an autonomous agent", *Transactions of the institute of system, control and information engineers*, Vol. 15, No. 7, pp. 350-358, 2002.

Multi-robot mutual localization using space-division infrared wireless communication

Hiroyuki Takai*, Takamichi Onishi** and Keihachiro Tachibana*

*Department of Information Machines and Interfaces, Faculty of Information Science
Hiroshima City University
{takai,tatibana}@sys.im.hiroshima-cu.ac.jp

** Department of Information Machines and Interfaces,
Graduate School of Information Sciences, Hiroshima City University
onishi@sys.im.hiroshima-cu.ac.jp

Abstract

Recently, multi-agent robot systems that can perform team operations such as exploration of hazardous environments have been developed. These robots help each other by sharing location information.

Normally, in hazardous environments, geo-navigation cannot be used, because landmarks are hard to prepare in advance. We propose a detection method of mutual location information for multi-agent robot systems. In this method, all robots that are communicating with each other provide the role of landmarks. These robots detect the angle of arrival (AOA) of communication signals and compute location information by triangulation.

We designed an optical wireless inter-robot communication system. In this system, several transceivers that detect the AOA are put on the circumference of the robot body and they face outwards. We tested the accuracy of the AOA detection and then tried computation of mutual location information using triangulation based on the AOA.

1. Introduction

In recent years, multiple autonomous mobile robots that can perform group operations have been developed [1]. These robots are used for planet investigations, deep ocean surveys, assistance to rescue operations, maintenance of nuclear reactors, and so on.

When the number of robots increases in the same working area to accomplish their tasks, they will often collide with adjacent robots.

Therefore, these robots have to communicate with adjacent robots to cooperate. Radio waves or infrared rays are used as carrier in inter-robot communication.

Usually, radio waves are used for the remote control of mobile robots. However, radio waves spread out omni-directionally and cause communication interference in unexpected places.
On the other hand, infrared rays have strong directivity and limited beam width. Accordingly, infrared communication has to track communicating partners to maintain communication links.
When the number of robots increases in the working area, omni-directional signals increase the communication interference. Consequently, infrared communication would be suitable for such a communication environment.

Location information and mutual communication is extremely useful to avoid collisions and to execute tasks smoothly. In hazardous environment, because landmarks are hard to prepare in advance, detection of global location is difficult. However, if robots spread out the communication networks in the working area, landmarks will not be necessary for the detection of mutual location information, because these robots can provide the role of landmarks.
Multi-robot mutual localization can be achieved by the triangulation that is based on angle of arrival (AOA) of communication signals from adjacent robots.

This paper describes the method of multi-robot mutual localization using inter-robot communication.

2. Inter-robot wireless communication using infrared rays

As previously mentioned, infrared communication would be suitable for mutual robot communication to reduce the communication interference encountered when radio waves are used. Infrared rays have strong directivity and limited beam width. Therefore, infrared communication easily loses the connection, when communicating robots move or rotate.

We designed an optical wireless communication system. In this system, several transceivers are put on the circumference of the robot body and they face outwards. Figure 1 shows the arrangement of transceivers. These transceivers detect AOA of communication signals to maintain communication links by switching to transceivers that face communicating partners.
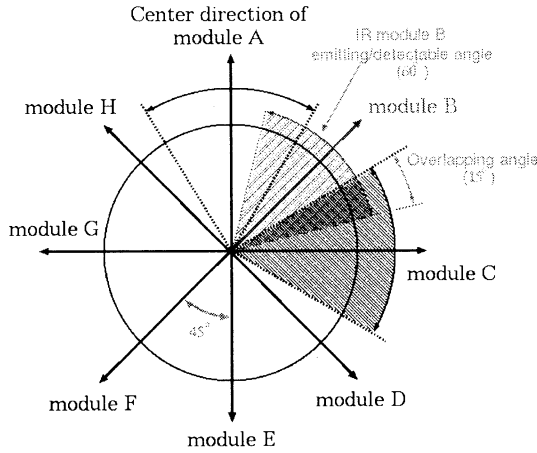
Fig.1 Arrangement of transceivers

Infrared communication suffers from hardly any interference, when an infrared receiver faces the different direction from transmission signals. Therefore, the communication system is able to communicate at the same time in parallel with different partners in different positions by using transceivers independently. Space-division communication can take place.

When each robot relays information between different partners, a communication network is created in the working area. It is an ad hoc communication network because each robot is independently mobile and may change position depending on the task.

Figure 2 shows an inter-robot communication network.
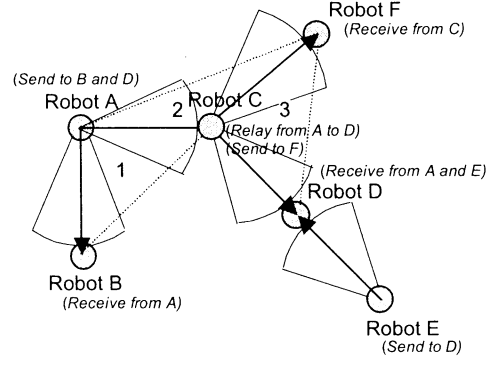
Fig.2 An inter-robot communication network

When a receiver catches signals from two or more partners at the same time, received signals are confused. On that occasion, these robots are making triangles. From the geometrical nature of triangles, if each reception angle is restricted to 60 degrees or less, a robot that has the biggest interior angle can communicate without confusion. Each interior angle computes from AOA of communication signals. Therefore, these robots exchange interior angle information and select the one that has the biggest interior angle to become an arbiter. After that the arbiter controls their transmission. Figure 3 shows a selection of an arbiter. When robots move and the triangle changes shape, the arbiter role is handed over to another robot.
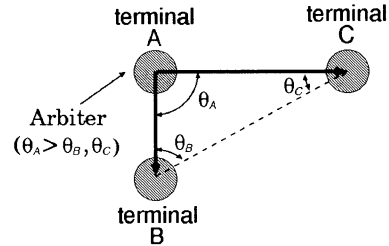
Fig.3 A selection of an arbiter

3. Multi-robot mutual localization using infrared wireless communication

Location information is extremely useful for multi-robot cooperative operations. Normally, geo-navigation is used for detecting global location information. However, in hazardous environments, geo-navigation is difficult to use for localization, because landmarks are hard to prepare in advance. These robots detect preferentially mutual location information rather than global location information to execute their tasks, because these robots help with adjacent robots.

Mutual location information computation is achieved by triangulation based on mutual robot communication. When these robots are communicating with each other, they can detect AOA of communication signals, and they provide the role of landmarks that is for triangulation. Mutual location information is computed in the following method from the AOA of communication signals.

Computation method of mutual location information requires three known landmarks in the working area. These landmarks are arranged on coordinates from $P_1(x_1,y_1)$ to $P_3(x_3,y_3)$, and P_2 is the origin. When, the coordinate of the robot is $P(x,y)$ and the movement direction of the robot is θ, these parameters are computed from equation 1.

$$\left. \begin{array}{l} x = \overline{p_2p} \cos \phi \\ y = \overline{p_2p} \sin \phi \\ \theta = \phi - \theta_{01} - \theta_{12} + \pi \end{array} \right\} \ (1)$$

Parameters are computed as follows.

$$\phi = \tan^{-1} \frac{\overline{p_1p_2} \sin(\theta_{12} + \alpha) \sin \theta_{23} - \overline{p_2p_3} \sin \theta_{12} \sin \theta_{23}}{\overline{p_1p_2} \cos(\theta_{12} + \alpha) \sin \theta_{23} + \overline{p_2p_3} \sin \theta_{12} \cos \theta_{23}}$$

$$\overline{p_2p} = \frac{\sin \theta_{23} \cos \phi + \cos \theta_{23} \sin \phi}{\sin \theta_{23}} \overline{p_2p_3}$$

$$\alpha = \tan^{-1} \frac{y_1}{x_1}$$
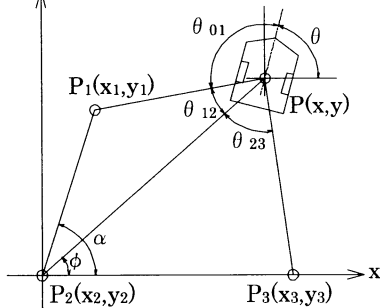
Fig.4 Localization using triangulation

Figure 4 shows localization using triangulation based on the AOA of communication signals. The robot detects AOA (θ_{01}, θ_{12}, θ_{23}) of communication signals and computes coordinate $P(x,y)$ by triangulation.

In hazardous environment, these landmarks will be set on the entrance of the working area or the supply base, because landmarks are hard to prepare in advance. After that, each robot computes its own location information and then creates a triangulation network.

4. Hardware design and performance measurements

This infrared communication system finds the direction of a communicating partner and gets location information about each robot. This process depends on the detection of the AOA of infrared signals. The PIN photo diode (HAMAMATSU S6560) is the detection device of the infrared rays in this experiment. Figure 5 shows a schematic view of the PIN photo diode.
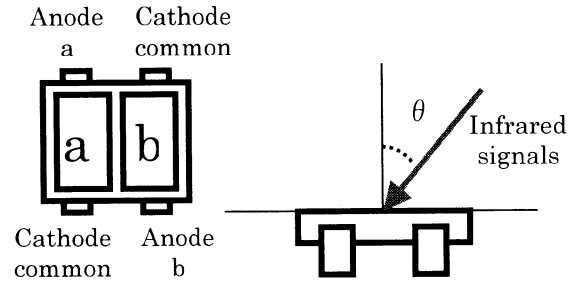
Fig.5 A schematic view of photo detector

The electric current output 'a' and 'b' of the PIN photo diode has the relations of the equation 2 with the AOA of the infrared signals θ.

$$\theta = (a - b) / (a + b) \qquad (2)$$

Figure 6 shows the angle detection error of this infrared detector.
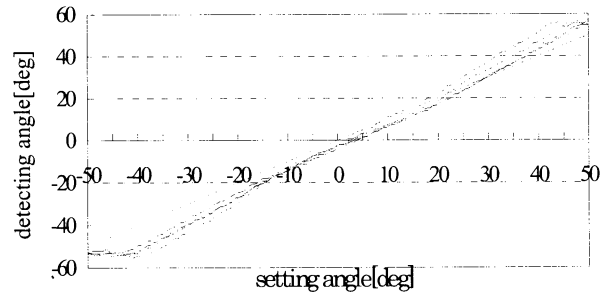
Fig.6 Angle detection error of detector

In this communication system, eight transceivers are arranged at 45 degrees intervals on the circumference on the robot body, as shown in figure 1. Each transceiver has a limited reception angle of 60 degrees overlapped by 15 degrees on each side by adjacent transceivers. Barriers that have the opening angles of 60 degrees are set in front of detectors and restrict reception areas.
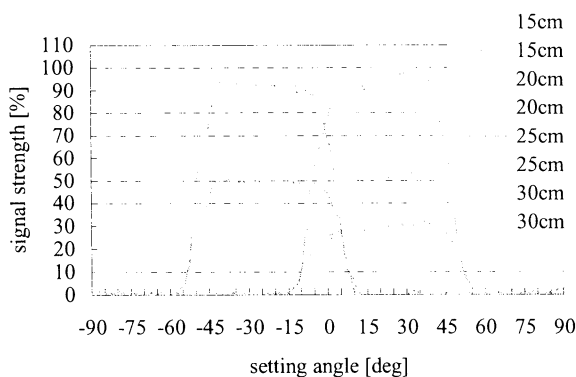
Fig.7 Reception areas of two detectors

Figure 7 shows the reception area of the two adjacent detectors. This result confirms the feasibility of the layout of figure 1.

The accuracy of location information computation was tested based on the detected AOA. When three landmarks ware arranged at apexes of an equilateral triangle and a robot moved inside the triangle, results of the localization had approximately 90% accuracy with the true value.

Conclusions and future works

We proposed a space-division infrared communication system for mobile robots in this paper. We verified the communication system that can separate receiving signals from different direction using different receivers.

In an experiment of detection of AOA, the result is able to detect an AOA precisely. The accuracy of localization is approximately 90% with the true value based on the detected AOA. Using statistical methods will reduce the AOA detection errors and the localization errors.
If the robot is placed where the AOA is a right angle, accurate location information cannot be computed by triangulation. In this system, robots can create communication networks. Therefore, robots will be able to find desirable landmarks to compute accurate location information.

For cooperative operation smoothly, each robot will share the working area map. The location information will be integrated with other sensor data such as ultrasonic sonar to make the working area map.

References

1. ASAMA H., et al (Eds.) (1994) "Distributed Autonomous Robotic Systems" (DARS), Springer, Tokyo

2. TAKITA Y, et al (1995) "Navigation and Recognition of the Unknown Space Using Infrared-Following Servo System for Moving Robots", Transactions of the Japan Society of Mechanical Engineers, vol.61, no. 590, series C, pp.209-216

3. TAKAI H., et al. (2001) "A Space-Division Optical Wireless Communication System for Fully Distributed Multiple Autonomous Mobile Robots". In: K.SCHILLING, H.ROTH (Eds.). Telematics Applications in Automation and Robotics 2001 (TA2001). PERGAMON press, pp.333-338

4. TAKAI H. et al. (2001) "A Space-Division Optical Wireless Inter-ROBOT Communication System with Mutual Localization Ability for Multiple Autonomous Mobile ROBOTS", In: INOUE H., ASAMA H. (Eds.). Preprints of The 4th IFAC Symposium on Intelligent Autonomous Vehicles (IAV2001) at Sapporo, Japan, Sep.5-7, 2001. IFAC, 338-343

5. TAKAI H. et al (2002) "A Geometric Arbiter Selection Algorithm on Infrared Wireless Inter-robot Communication", In: ASAMA H., ARAI T., FUKUDA T., HASEGAWA T (Eds.). Distributed Autonomous Robotic Systems 5 (DARS 5), Springer, Tokyo, pp.61-70

Effects of Information Sharing on Collective Behaviors in Competitive Populations

Reiji SUZUKI and Takaya ARITA

Graduate School of Human Informatics, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan
{reiji, ari}@info.human.nagoya-u.ac.jp

Abstract

This paper explores the effects of information sharing on diversity and adaptivity of behaviors in competitive populations. We focus on the information supplement of congestion status at the events or theme parks. We have conducted a multi-agent simulation, in which each agent visiting the attractions is able to avoid the crowded ones by using the shared information among agents with a certain probability. Experiments have shown that there exists the optimal probability that minimizes the congestion deviation among attractions. We also discuss the cause of a sudden decrease in this effect of information sharing, which has been observed in long-term experiments.

Keywords: information sharing, behavioral diversity, collective behavior, El-farol bar problem, Minority Game.

1 Introduction

The origin and maintenance of diversity have been explored in various fields concerning complex systems. Especially, behavioral diversity in social populations of agents has been discussed mainly in social and economic sciences, because it is one of the essential factors that determine the dynamics and adaptivity of competitive populations.

Arthur devised the El-farol bar problem [1] to study a competitive interaction of boundedly rational agents, in which each agent faces the binary choice of either attending or not a bar which is enjoyable only if it is not too crowded. He had shown that inductive reasoning of agents yielded a diversity of behavior among agents and made the average attendance self-organize to around the desirable capacity of the bar. Challet and Zhang also proposed a simple version of Arthur's model, which is known as the Minority Game [2]: N agents have to choose an action from 0 or 1. Those agents who have chosen the minority win, the others lose. Each agent possesses a finite set of strategies which are represented as mappings from possible M-length history of minorities' choice to its next action, and uses the one which would have been the most rewarding if it had been used since the beginning. Savit, Manuca and Riolo have found an environmental condition ($\rho = 2^M/N$) that minimizes the volatility (the standard deviation of the number of winners) which is smaller than that in the case when all agents randomly choose actions [3].

Diversity of information shared among agents is also an essential factor because it can directly or indirectly affect the diversity of collective behaviors. Hogg and Huberman constructed an abstract model which is composed of interacting agents choosing resources based on imperfect or delayed information about rewards which depend on the frequency of agents' choices [4]. In their model, the population dynamics becomes chaotic as the delays and the uncertainty of information increase. But if each agent has an additional delay or imperfection about information that are genetically defined, the population evolves to a heterogeneous genetic composition, and the chaotic behavior is completely freezed out. Akaishi and Arita focused on adaptive property of misperception that can increase the diversity of information that facilitates flocking toward a limited resource in a foraging task [5]. They had shown that diversity of information, which was increased by misperception, could also increase the diversity of behavior and adaptivity of population. These studies imply that sharing with the same information among individuals does not always yield a desirable result for whole population.

The purpose of this study is to explore the direct or indirect effects of information sharing on diversity and adaptivity of behaviors in competitive populations by focusing on the information supplement of congestion status at events or theme parks. In particular, we aim at clarifying the dynamic properties of these effects so as to understand them in more realistic situations. We have constructed a model in which each agent visiting the attractions is able to avoid the crowded ones by

using the shared information among agents with a certain probability. By altering the probability of information sharing in our model, we consider the correlation between informational and behavioral diversities.

2　The Model

There exist N_a attractions on the event space that is represented by two-dimensional $(W_x \times W_y)$ grid as shown in Figure 1. There are N_p agents (visitors) and at most N_e agents can enter the space from the entrance every time step with the restricted admission that the total number of agents in the event space should not exceed the maximum number N_m. Initially, each agent decides whether it will visit each attraction $A_i(i = 0, \cdots, N_a-1)$ or not with a probability P_i respectively. P_i represents the popularity of each attraction. Each agent also chooses one from them randomly as a first destination to visit.

In every time step, the order of each agent's turn for action are randomly decided, and each agent can execute an action in its turn based on its current mode: *walking, waiting* and *experiencing*. In *walking* mode (initial mode for all agents), each agent can walk toward its destination with the probability $1-(n+1)/10$, where n is the number of neighboring cells within a 3 by 3 window that are already decided as next position of other agents or outside of the event space, except for attractions. This factor stands for the interference among agents caused by congestion. If the agent can walk, the x and y coordinates of the agent are updated by the following equations:

$$x \leftarrow x + rsgn(A_{i_x} - x), \qquad (1)$$

$$y \leftarrow y + rsgn(A_{i_y} - y), \qquad (2)$$

where A_{i_x} and A_{i_y} denote the x and y coordinates of the destination A_i. $rsgn(k)$ is a fuction which returns a randomly chosen value from $-1, 0$, or 1 with probability p_r; otherwise returns $-1, 0$, or 1 if $k < 0, k = 0$, or $k > 0$ respectively. If the agent is going to take an oblique direction, it is allowed to walk with an additional probability $2^{-0.5}$, which makes an average walking speed per step independent of its direction.

Agents who have arrived at their destinations enter into *waiting* mode, and wait for their turns at the tails of waiting lines until they have their turn. It is because that each attraction A_i is capable of giving entertainment at most C_i agents at once. Agents are in *experiencing* mode while they are going through the programs for L_i time steps. When an agent has finished going through the program there, it chooses next destination from its remaining attractions to visit
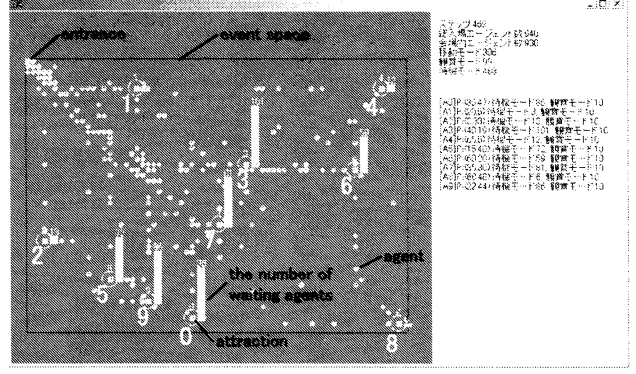
Figure 1: The screenshot of the simulation.

using the information about congestion status of all attractions, and then enters into *walking* mode again. The information is provided for each agent with a certain probability r through an information equipment such as PDA. If an agent receives the information, it heads for the attraction with the shortest waiting line at present among its remaining attractions to visit; otherwise it heads for the nearest one from its current position in slant distance. The probability of information sharing r is the parameter which decides the informational diversity among agents in our model. The agent who has no remaining attractions to visit is eliminated from the event space. A simulation ends when all agents have been eliminated.

3　Experimental Results

3.1　Effects of Information Sharing on Behavioral Diversity

We arranged each attraction $A_i(i = 0, \cdots, 9)$ as shown in Figure 1 and adopted the same parameters for all attractions: $P_i = 0.7$, $L_i = 30$ and $C_i = 10$. We also used the following values as the other environmental parameters: $W_x = 70$, $W_y = 50$, $N_a = 10$, $N_p = 2000$, $N_e = 2$, $N_m = 1000$ and $p_r = 0.05$.

Firstly, we clarify the basic behavior of population caused by the environmental condition (arrangement of attractions). Figure 2 shows a typical trial in case of $r = 0$, in other words, no information is shared among agents. Horizontal axis shows the time step and each line shows the number of waiting agents (agents in *waiting* mode) at each attraction (the line "Ax" corresponds to the attraction x in Figure 1). In this paper, we consider behavioral diversity of whole population by focusing on these values.
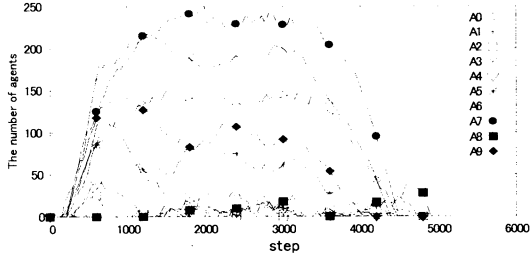
Figure 2: The number of waiting agents at each attraction ($r = 0.0$).
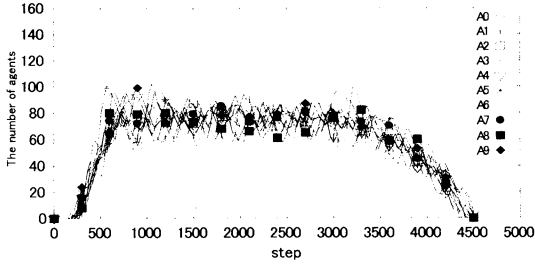
Figure 3: The number of waiting agents at each attraction ($r = 0.5$).

Figure 4: The number of waiting agents at each attraction ($r = 1.0$).
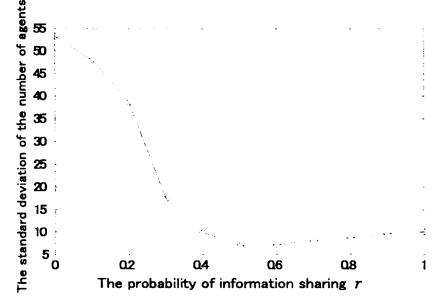
Figure 5: Correlation between r and the standard deviation of the number of waiting agents among attractions.

From the initial condition, many agents entered into the event space and headed for their destinations, then the number of waiting agents increased at all attractions. But there were large differences among their peaks through the trial. The centrally-located attractions on the event space (such as attraction 3 and 7) had many waiting agents. In contrast, peripherally-located attractions (such as attraction 1 and 8) had few waiting agents. The centrally-located attractions are close to many other attractions and they are preferentially selected by agents as next destinations, because each agent always chooses the nearest one from the remaining attractions to visit when $r = 0$. Thus, the centrally-located attractions tend to be crowded with agents. It means that the behavioral diversity among agents is decreased by the environmental condition in the sense that agents have an inclination to concentrate into certain attractions.

Next, we have conducted experiments using various probabilities of information sharing r. Figure 3 and 4 depicts the examples in case of $r = 0.5, 1.0$, and Figure 5 shows the standard deviation of the number of the waiting agents among attractions in various cases of r. As r increased, the transitions of the number of waiting agents dramatically got closer as shown in Figure

3, and then the deviation reached the minimum value when $r = 0.5$. However, as r increased, the deviation slightly increased when $r \geq 0.6$. Figure 4 ($r = 1.0$) tells us that it is caused by the large fluctuations in the number of waiting agents compared with Figure 3 ($r = 0.5$). This is due to the fact that sharing of the same information among too many agents rather causes new concentration of agents into another attraction that was not crowded a little time ago. Thus there exists an optimal informational diversity that maximizes the behavioral diversity that was basically decreased by the environmental condition.

3.2 A Dynamic Shift in the Effect of Information Sharing

Subsequently, we have conducted experiments using the condition: $r = 0.5, N_p = 20000$, so as to clarify how the effects of information sharing shown in previous section changes through long-term experiments. We have observed the dynamic shift in the effect of information sharing in about 50 percent out of 30 trials in this setting. Figure 6 shows a typical example of such a phenomenon. We can observe the sudden
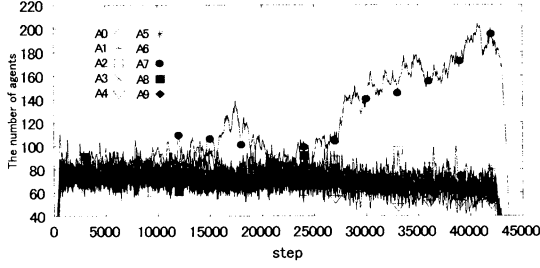
Figure 6: The number of waiting agents at each attraction ($r = 0.5, N_p = 20000$).

Figure 7: The number of agents who are due to visit each attraction ($r = 0.5$).

increase in the number of waiting agents at the attraction 7, from step 15000 to 20000 and after 25000.

The reason for such a decrease in the effect of information sharing is explained by the side effect of information sharing on future behavioral diversity. Agents who have received the information tend not to choice the centrally-located attractions such as the attraction 7. Thus, the deviation of the waiting agents among attractions was kept small in early period of the trial. However, at the same time, it gives rise to the gradual decrease in future behavioral diversity: the increase in the deviation of the number of agents who are due to visit each attraction (who have each attraction in their remaining attractions to visit) in the event space. Figure 7 depicts the number of agents who are due to visit each attraction in the same trial as Figure 6. Since the agents' destinations are chosen from their remaining attractions to visit, the deviation of these values reflects on the future behavioral diversity among agents. In fact, the number of agents who were due to visit the attraction 7 relatively got larger as time elapsed. This means that the future behavioral diversity was gradually decreased in the sense that more agents were due to visit the attraction 7 than other attractions. This decrease in the future behavioral diversity tends to make agents choose the attraction 7 even when they receive the information (especially in case where they have only the attraction 7 in their remaining attraction to visit). Then the concentration of agents into the attraction 7 occurred at last.

4 Conclusion

We have discussed how information sharing affects diversity and adaptivity of behaviors in competitive populations by focusing on the information supplement of congestion status at events or theme parks. By conducting multi-agent simulations, the following
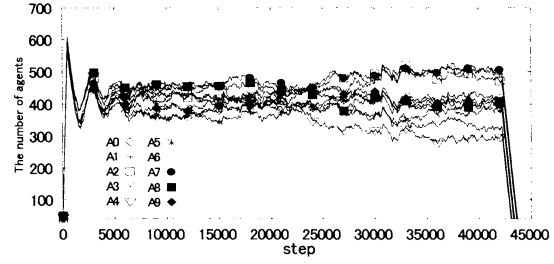
results were obtained: First, the behavioral diversity among agents is decreased by the environmental condition such as the arrangement of attractions on the event space. Second, there exists an optimal probability of information sharing that minimizes the standard deviation of congestion degree among attractions. Third, the dynamic shift in this effect in long-term experiments, the sudden increase of the waiting agents at a certain attraction, is caused by the side effect of information sharing, in other words, the gradual decrease in the future behavioral diversity among agents.

We believe that this series of study on the effects of information sharing would not only deepen our understanding of the collective behavior in human society but also could develop a new design principle for multi-agent robotic systems with a population of robots having complex interaction among them.

References

[1] W. B. Arthur, "Inductive Reasoning and Bounded Rationality," *American Economic Association Papers Proceedings*, Vol. 84, No. 2, pp. 406–411, 1994.

[2] D. Challet and Y. C. Zhang, "Emergence of Cooperation and Organization in an Evolutionary Game," *Physica A*, Vol. 246, pp. 407–418, 1997.

[3] R. Savit, R. Manuca and R. Riolo, "Adaptive Competition, Market Efficiency, and Phase Transitions," *Physical Review Letter*, Vol. 82, No. 10, pp. 2203–2206, 1999.

[4] T. Hogg and B. A. Huberman, "Controlling Chaos in Distributed Systems," *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 21, No. 6, pp. 1325–1332, 1991.

[5] J. Akaishi and T. Arita, "Misperception, Communication and Diversity", *Proceedings of Artificial Life VIII*, 2002 (in press).

Identification and learning of other's action strategies in cooperative task

Shuji Tohyama[†] Takashi Omori[†] Natsuki Oka[‡] Koji Morikawa[‡]
† Graduate School of Engineering, Hokkaido Univ.
Kita13jyou Nishi8chome, Kita, Sapporo, Hokkaido, 060-8628, Japan
tohyama, omori@complex.eng.hokudai.ac.jp
‡ Humanware Technology Research Laboratory,Matsushita Electric Industrial Co., Ltd.
3-4, Hikaridai, Seika, Soraku, Kyoto, 619-0237, Japan
oka@mrit.mei.co.jp, morikawa@crl.mei.co.jp

Abstract

During cooperative behavior with other people, we can coordinate our own actions to adapt to their behavior. To behave in this way, we need to understand the other's intention and goal through the observation of their behavior and the estimation of their state by using a mechanism such as the "other's model". In this paper, we propose a model of a learning agent that changes its own or "self" actions through observations and predictions of other agents' actions in a cooperative task in a multi-agent environment. In order for the learning agent to learn prediction of another agent's actions, we made several functions to represent the "other's model" and mechanisms to switch these functions to the learning agent, based on the model proposed by Nagayuki et al (2000). In our simulation, the learning agent acquired several "other's models" and selected the appropriate one. Moreover, the agent changed its behavior by using the "other's model" when the other agent changed identity and strategy.

keyword : intention, Q-learning, cooperation, other's model

1 introduction

In a cooperative situation, humans are able to change their behavior appropriately by adjusting to other people's behavior. When we perform a cooperative task with several people, such as carrying a long table, we should observe the behavior of the other people or communicate with them. If, for example, one person moves to one end of the table, another person should move to the other end. In such situations, we estimate the other person's intention and determine our own behavior (i.e. which end we should move to) just by observing the other persons behavior.

In this study, we propose a computational model of another agent's intentions and simulate how an agent acquires the "other's model" through several cooperative tasks in a multi-agent environment. In the experimental simulation, the agent acquires several "other's models" which are dependent on the other agent's probable intentions and selects the most appropriate one in order to determine the intention of the other agent by observing his/her behavior. The agent then acts based on the prediction of the other agent's strategies derived from the model. Next, we analyze the agent's learning mechanisms for acquiring the "other's model" and the flexibility of the agent's decision making in terms of its behavior in response to the other agent's strategy.

2 Model for estimating other agent's intention

2.1 Estimation of intention

Here, we propose a model of the human cognitive process in a cooperative task, which has several goals as follows: a human observes and evaluates another's behavior to determine what his behavior should be to achieve the goal of the cooperative task. Looking at this process from the perspective of information processing, we can say that the agent observes the other agent's behavior and calculates its likelihood using the "other's model". This model is defined as a probability of action a in the current state s when the "other" has a different intention. If the likelihood is high enough, the agent judges that the observed agent is "identical" with the "other's model". Moreover, we propose that the agent estimates the other agent's intention depending on its chosen "other's model". The agent calculates a likelihood of an observed sequence of ac-
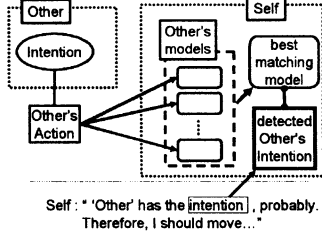
Figure 1: estimation of other agent's intention.

Figure 2: estimation of other agent's action.

tions using its "other's model" and selects the most suitable "other's model". Thereafter, the agent judges the intention of the other agent by using the selected "other's model" (Fig. 1).

2.2 Other's model

We used a model proposed by Nagayuki et al [1] as a mechanism to predict another agent's action. In that model, an agent has a total Q-value function $Q(s, a_s, a_o)$, where the variable s represents current state, a_s self action and a_o other's action, as an extension of Q-Learning[2]. Among the variables, the learning agent must predict the other agent's action a_o with a function $I(s, a_o)$, which is called the "model function of another agent" because the agent cannot know the exact value of a_o. When the agent has observed the actual act (a^*) the other agent on a particular state s, the $I(s, a_o)$ function can be recalculated as follows:

$$I(s, a_o) = (1 - \theta)I(s, a_o) + \begin{cases} \theta & (a_o = a_o^*) \\ 0 & (a_o = otherwise) \end{cases}$$

The learning process of "other's model" is as follows:

1.The agent estimates a self Q-value function $Q(s, a_s)$ from the estimated other agent's action a_o derived from $I(s, a_o)$ on the current state s. Then, a self action a_s is chosen based on the stochastic policy π:

$$\bar{Q}(s, a_s) \equiv \langle Q(s, a_s, a_o) \rangle_{I(s, a_o)}$$

$$= \sum_{a_o \in A_o} I(s, a_o)Q(s, a_s, a_o),$$

$$\pi[a_i | s] = \frac{e^{\bar{Q}(s, a_i)/\tau}}{\sum_{a_k \in A_s} e^{\bar{Q}(s, a_k)/\tau}}$$

2.The agent executes the action a_s and the other agent executes the action a_o simultaneously. Thereafter, the environment changes to a novel state s' and the agent is given a reward r from the environment. Then, the agents update $I(s, a_o)$ depending on

the equation described above and the total Q-function $Q(s, a_s, a_o)$ as follows:

$$Q(s, a_s, a_o^*) \leftarrow (1 - \alpha)Q(s, a_s, a_o^*)$$

$$+\alpha(r + \gamma \max_{a_s' \in A_s} Q(s', a_s', a_o')),$$

where $a_o' = \arg\max_{a_o' \in A_o} I(s', a_o')$.

3.The agents update s to s', and then repeat steps 1 and 2 until the state s satisfies a terminal condition.

2.3 Selection of other's model

In this experimental condition, we designed the other agent so that the action strategy would change suddenly. Therefore, the agent with the "other's model" can identify the change in action strategy by using the model function of the other's $I(s, a_o)$ depending on observed action a_o. The model function $I(s, a_o)$ should have a higher value if the other agent has estimated correctly. For example, when the other agent approaches a prey from the right side, then the self agent should catch the prey from the left side by looking at the behavior of the already known strategy of the other agent (Fig. 2). In this case, the next step action a_o^* can be predicted with a high value of $I(s, a_o)$. On the other hand, the model function $I(s, a_o)$ has a lower value if the other agent intends to approach from the upper side.

To adapt to the sudden changes in intention of the other agent, the self agent must have a mechanism for selecting the appropriate other's model $I(s, a_o)$ from multiple other's models by calculating likelihood I_M using a memorized other agent's action sequence consisting of the last 75 time steps:

$$M = \arg\max_{m \in having model} \sum_{t=0}^{-74} \log I_m(s(t), a_o(t))$$

: hunter ▲ : prey
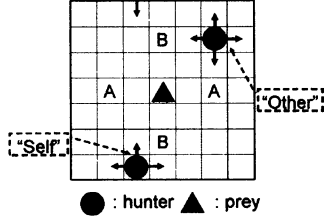
Figure 3: learning environment.

where $s(t)$ is the state in time t, $a_o(t)$ is the action of other's in time t.

The self agent uses the $I_M(s, a_o)$ as the model of the other agent's action if the likelihood M is higher than a defined segmentation threshold. If that is not the case, the self agent makes a novel "other model function" (I'_M) from the current $I_M(s, a_o)$ and learns for a certain period until the new $I_M(s, a_o)$ becomes stable enough. It is assumed that the other's agent's strategy does not change during the learning period.

3 Experiment

3.1 Environment

To evaluate the performance of the model in a co-operative task, we prepared a variation of the pursuit problem [3]. In a 7×7 toroidal grid world, two hunter agents catch one prey (Fig. 3). The hunters simultaneously perform one of five actions, such as staying at, or moving up, down, left, or right from the current position at a discrete time step.,The prey does not move. The goal of this task is for the hunter agents to capture the prey. The hunter agents are placed at random positions on the grid world and are able to capture the prey only when both agents approach the prey simultaneously from point A or B on the grid world of Fig. 3. When the agents capture the prey, one episode ends and the agents each receive a reward. We represented the current state s of the agents using a combination of relational positions between the self agent and the prey and between the other agent and the prey. In Fig. 3, s is represented as $([1, 3], [-2, -2])$.

3.2 Task

We divided this experiment into two phases.

First phase : We made four kinds of other agents, that had learned to approach the prey from up-side (up agent), down-side (down agent), left-side (left agent) or right-side (right agent). Each agent learned

its particular strategy by receiving a reward only on achieving its goal state. The self agent acquired four kinds of other model functions, $I_m(s, a_o)(m = 1...4)$, and the total Q-function $Q(s, a_s, a_o)$ during 3000 episodes of initial learning. Thereafter, a new Q-function, whose initial value was a mean value of the four different Q-functions, and the other's model function $I_m(s, a_o)$ with a flat initial value were generated. The self agent had the integrated Q-function and received one "other's model" at the beginning of the next phase.

Second phase : The self agent experienced the first 3000 episodes with the up agent, the next 5000 episodes with the down agent, the succeeding 5000 episodes with the left agent and the final 5000 episodes with the right agent, with no notification of the change to a different agent. From this, we evaluated whether the self agent acquired a satisfactory "other's model". During these 18000 episodes, we inserted 40 evaluating episodes after every 10 learning episodes in which the self agent tried to capture the prey with each of the four other agents and we counted the time steps taken to achieve the goal in each case. The self agent was only able to change the other's model function, but did not update these functions in the evaluating episodes. We developed three kinds of self agents in order to evaluate the self agent's ability to estimate the other agent's action. One was a self agent that had a mechanism for the selection and segmentation of other's model functions (Self A). The second self agent had no other's model function (Self B) and the third self agent had just one other's model function (Self C).

The learning parameters were $\alpha = 0.3, \gamma = 0.9$ in both phases. The value of $\tau = 0.2 \times 0.998892^{numep}$ in the first phase (where $numep$ was the number of learning episodes and tau was initialized to 0.2 at the beginning of the learning episodes with a novel other agent). The value of τ was fixed to 0.05 in the second phase. The segmentation threshold for making the novel other's model function was -35.0 and the number of learning periods after the making of a novel other's model function was 3000. The initial value of total Q-function was zero and the initial value of the other's model function was 0.2.

3.3 Result

In Fig. 4, we show the number of averaged time steps during the evaluation episodes from 20 simulation examples. Comparing Self A with Self C, there was no difference in performance during the first 3000 episodes. However, Self C had greater time steps than
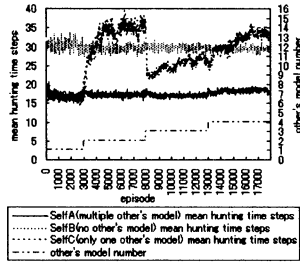
Figure 4: mean hunting times and number of other agent models.
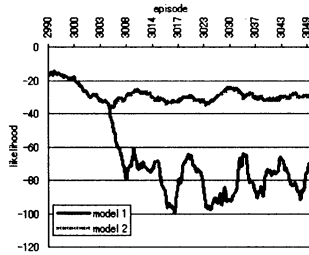
Figure 5: likelihood for down agent from episode 2990 to 3050. Other's model is divided into old one and a novel one at episode 3004.

Self A when a novel other agent appeared, whereas for Self A there was not a great difference in the number of time steps. Moreover, Self A segmented novel other model functions at the appropriate time when novel other agents appeared at the 3000th, 8000th and 13000th episode in the second phase (Fig. 4). We show that, for Self A, the transitions of likelihood are dependent on the emergence of novel other agents: in Fig. 5 (up agent to down agent), Fig. 6 (down agent to left agent) and Fig. 7 (left agent to right agent). The Self A agent selected a suitable "other's model" evaluating the other agents as soon as they appeared, decreased the values of likelihood of another "other's model" below the segmentation threshold, and was able to estimate the intentions of the other agents correctly.

4 Discussion and Conclusion

We proposed a computational model in which the learning or self agent evaluates another agents intention by using its internal "other's models" which depend on its own observation of the other agent's action at a particular state. Furthermore, we confirmed that the self agent consistently made an appropriate novel
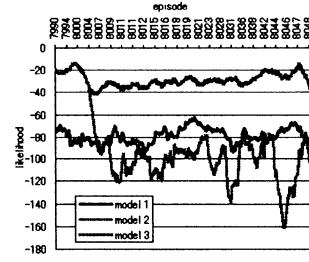
Figure 6: likelihood for left agent from episode 7990 to 8050. Other's model is divided into old one and a novel one at 8004.
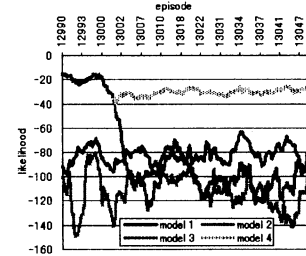
Figure 7: likelihood for right agent from episode 12990 to 13050. Other's model is divided into old one and a novel one at 13002.

"other's model" when the current other's action strategy was changed to a novel one without notice.

In this paper, the self agent needed long episodes to learn a new "other's model" and it observed only four other agents. That the agent should learn "other's models" more quickly and adapt to the actions of many of other agents are future problems which need to be addressed.

References

[1] Y.Nagayuki, S.Ishii and K.Doya, "Multi-agent reinforcement learning: an approach based on the other agent's internal model," *Fourth International Conference on MultiAgent Systems (ICMAS)*, pp. 215-221, Los Alamitos: IEEE Computer Society (2000).

[2] C. J. C. H. Watkins, P.Dayan, "Q-learning," *Machine Learning*, 8, pp. 279-292, 1992.

[3] M.Benda, V.Jagannathan, and R. Dodhiawalla, "On optimal cooperation of knowledge sources," *Technical Report*, BCS-G2010-28, Boeing AI Center, 1985.

Matching with Feature Segments of Stereo Images by IA

Hiroki Kakiuchi
Dept. of Electrical and Electronic Eng.
Fukui University
3-9-1 Bunkyo, Fukui 910-8507, Japan

Kozo Okazaki
Dept. of Electrical and Electronic Eng.
Fukui University
3-9-1 Bunkyo, Fukui 910-8507, Japan

Abstract

Matching of stereo images is a fundamental task for
3D recovery. We have matched with feature points of
a direct image and its mirror image by using GA and
the information of apexes connections. In this paper,
we propose a matching algorithm of feature segments
by using cut set graph and IA. First, the stereo images
are specified to a standard and reference image, respec-
tively. The feature segments of standard image are
classified to 3 groups; left (inner), cut set (center) and
right (outside) group. An affinity of reference image is
defined how many the remark segments are included
in each groups correctly correspond to the standard
groups. Here, in spite of high fitness, miss matching
is occurred because of graph imbedding problem. We
introduce an IA partly in IA to cope with these multi-
peaks of fitness cases.

keywords: matching, stereo images, IA, affinity, an-
tibody, cut set

1 Introduction

We have proposed a method of 3D recovery by us-
ing a direct image and mirror images, where we uti-
lized the plane symmetry of images [1]. Here, first the
apexes and line segments of images are extracted to
estimate a vanishing point (VP). Next, the apexes are
matched with each other using mirror constraints and
3D has recovered. However, the more the number of
the apexes and line segments increase, we copy with
the selection of apexes to estimate VP. The matching
points are known before hand, 3D recovery is easily
done.

In this paper, we propose a matching algorithm of
two stereo camera images, which we search for the or-
ders of based image apexes matching with those of ref-
erence image (MBR) by using immune algorithm (IA).
Here, we avoid death genes production. Travel sales-
man problem (TSP) by GA is similar to the above

orders search problem. TSP searches the order of
visiting cities how the integrated distance be mini-
mized. The different points between TSP and MBR
are as follows: (1) inverse order is error, (2) there
exist connect/non-connect constraints among apexes
and occlusions in images, (3) even the approximation
orders of cities is one of the solution for TSP, but
the solution of MBR is one and only one. Besides, a
similar figure, symmetry and rotational symmetry are
the same maximum valuation in MBR, (4) there exist
plural maximum peeks of adaptation value in case of
graph embedding (i.e., same connection matrix).

IA generate many variety of antibodies by antibody
manifold production mechanism and self-adjustment
mechanism. Therefore, we are possible to search the
optimum solution globally by avoiding local maxi-
mum.

2 Matching with feature segments of stereo images by IA

Immune means "get rid of disease", ex. contagious.
When antigens irrupt into a living body, the clones of
lymph that is fitting best to the antigens are produced
selectively. They remove the antigens and protect the
living body. They have a memory mechanism to the
antigen. The lymph or the antibody of IA regards as
" a solution to a task " . IA utilized the flame work of
immune mechanism.

2.1 Immune Algorithm (IA)[2]

Comparing IA to the search of optimum problems'
solution, the antigen coresponds to the constraints and
the purpose function, the antibody coresponds to op-
timum solution and the affinity between the antigen
and the antibody coresponds to the evaluation of the
solution respectively. The algorithm of IA is as fol-
lows:
(Step 1) recognition of antigen

The system recognizes the antigen as input information.

(Step 2) generation of initial antibody group

We generate antibody groups from memory cells, which were effective to antigens in the past. Practically, we read the effective antibodies (candidate of solution) in the saved database. Here, we generate them randomly in initial condition.

(Step 3) calculation of affinity

We calculate an affinity ax_v between an antigen and an antibody v and an affinity $ay_{v,w}$ between the antibody v and an antibody w. When ax_v is "1", the antibody succeeds to remove the antigen.

First, we calculate a connection strength opt_v between an antigen and the antibody v.The opt_v is defined as matching rate of the apexes and the line segments of the base image and those of reference image, namely,

· the line segments matching in the three groups of the cut set matrix.

The affinity ax_v between an antigen and the antibody v is defined as Eq.(1).

$$ax_v = \frac{1}{1 - opt_v} \tag{1}$$

(Step 4) specialization to memory cell

We add the antibody with high affinity obtained from (Step 3) to memory cells. The size of memory cell is limited the lowest antibody is removed. Practically, the affinities among the antibody and those of data are calculated by Eq.(2), and the highest one is exchanged.

$$ay_{v,w} = \frac{1}{1 + H(2)} \tag{2}$$

(Step 5) promotion and suppression of antibody production

An information entropy $H_j(N)$ of a gene coordinates of a antibody is expressed by Eq.(3),

$$H_j(N) = \sum_{j=1}^{S} -p_{i,j} \log p_{i,j} \tag{3}$$

here, $p_{i,j}$ is the probably of which i-th symbol appears at j gene coordinate.

$$p_{i,j} = $$

$$\frac{\text{Total number of } i\text{-th symbol appears at } j \text{ gene coordinate}}{N} \tag{4}$$
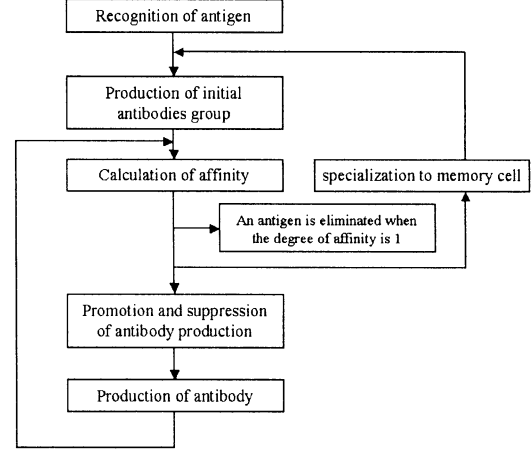
Figure 1: Immune Algorithm (IA)

Therefore, average information entropy of antibody gene $H(N)$ in the immune system is defined as Eq.(5),

$$H(N) = \frac{1}{M} \sum_{j=1}^{M} H_j(N) \tag{5}$$

and we keep the variety of the immune system.

Eq.(1) and Eq.(2) mean that the higher the affinity between an antibody and the antibody and the density, the higher the probability of survive in next generation. Therefore, Eq.(2) has two functions to adjust the immune mechanics variety; (1)promotes production of antibody with high affinity, (2) suppresses production of antibody with high density.

(Step 6) production of antibody

First, instead of removed antibody, a new antibody of which gene is randomly determined is produced. Next, this antibody are made crossover between one of the antibodies that is selected randomly among the remaining antibodies in (Step 5) and a new antibody multiplies. In addition, its gene is made change in certain probability by mutation process and we return back to (Step 3). We can product an antibody which respond to infinity number of antigen by this produce. Flow chart of IA is shown in Figure 1.

2.2 Coding of IA

We adopts Grefenstette algorithm[3], which generates no death gene for the coding of IA. When we multiply the antibody, we select two antibodies and their genes' coordinate randomly and make crossover. The

method does not succeed parents' genes, and the evolution is slow. Ex[4], Exx, and Sxx etc are proposed to succeed the parent's genes and promote the evolution for the case of TSP. However, it is possible to connect every city arbitrarily. MBRs' connections in this report are limited to the contrary. In addition, we cannot mark all apexes with one stroke generally. It is difficult to apply them to the matching.

First, we prepare a list $Y = (c_1, \cdots, c_n)$ of which order is arranged from B_1 to B_n arbitrary. Now, we code the order of $B^l = (b_1, b_2, \cdots, b_n)$. We express how the i-th line segment c_i corresponds to the order of list B^l assign to k_i. Here, k_i is integer ranged $1 <= k_i <= n - i + 1$. A list $K^l = (k_1, \cdots, k_n)$ in which the line segment is described as k_i is coupled with list $B^l = (b_1, \cdots, b_n)$ and we implement IA. All the genes of list K^l are made correspondence to those of B^l as one to one. We adopt mutation operation as the exchange of two genes of which coordinate is chosen randomly. When the operation is done to K^l, the order of list B^l shift all after some order. When the operation is done to B^l directly, only the two segments are exchanged and all the remaining segments are no-changed, i.e., we operate two points crossover to list K^l for the case of crossover and we only operate the exchange to list B^l for the case of mutation. The gene of list K^l is exchanged with that of list B^l is done. The inverse is the same.

3 Apexes matching by graph search

We regard the connection information's of segments and apexes obtained from G1 and G2 as an antigen and we evolve groups of antibody which gene are apexes order to be maximum affinity.

3.1 Connection matrix and cut set matrix

Suppose n is the number of apexes in base image, the number of branch in G1 tree is $n - 1$ and the cut set number is $n - 1$. We assign independent cut set C_i to be Eq.(6),

$$C_i = \sum_{k=1}^{n-1}(k), \quad (i = 1, \cdots, n - 1) \tag{6}$$

here, $C_i = {}_A C_i + {}_B C_i$ is the apex number and affix A and B of C_i correspond to G1 and G2 respectively.

v antibody's opt_v is the evaluation number how the features of reference image match with those of base
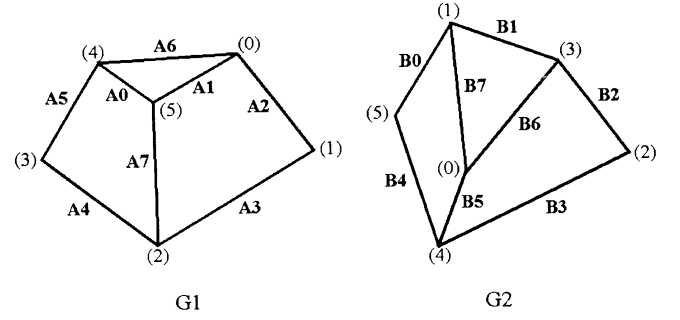
Figure 2: Ex. (G1:base image, G2:reference image)

image. The algorithm of opt_v calculation is as follows;
(Step 1)

Obtain the connection matrixes of G1 and G2 from data (shown in second team of Eq.(13) and Eq.(19), respectively).
(Step 2)

Determine non-singular matrix of G1 and G2 so as the level of apexes increasing order (shown in first team of Eq.(13) and Eq.(19), respectively).
(Step 3)

Calculate cut set matrix (mod 2). All that the element of cut set matrixes equal " 1 " correspond to the direct segment-cut (Eq.(13) and Eq.(19)).
(Step 4)

In addition, we group the features of graphs to three sets $S_i^{\langle * \rangle}$. Using connection cut set matrix $S_i^{\langle * \rangle}$ is defined as follows;

⟨1⟩ cut set $\equiv S_i^{\langle 1 \rangle}$

⟨2⟩ inner part set of cut set matrix $\equiv S_i^{\langle 2 \rangle}$

⟨3⟩ outer part set of cut set matrix $\equiv S_i^{\langle 3 \rangle}$

The procedure is as follows; First, we turn the elements of cut set matrix (0, 1) and pick up the non cut segments. Next, we judge them whether each of them belong to inner part or cuter part, respectively.

3.2 Verification

We verify whether the G2's cut segments corresponded to those of G1 are cut. The inverse verification (G2 to G1) is the same procedure. Namely, the cut segment in G1 is whether connect between the matched apexes of G2.

3.3 Matching

The task of matching search for the maximum f,

$$f = max \sum_{i=1}^{n-1} C_i \qquad (7)$$

For example, if the matching is correct, f is 0. In the case of occlusions of segments/apexes or anything features missing graph, $f = \sum C_i < 0$.

[Ex.] Cut set C_3 {(0),(1),(2) } evaluation opt_v

Ex. opt_v calculation for cut set C_3 {(0),(1),(2) } in Figure 2. Suppose an apex order of G1 to be A,

$$A[0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5] \qquad (8)$$

the correct apex order of G2 in B,

$$B[3 \quad 2 \quad 4 \quad 5 \quad 1 \quad 0] \qquad (9)$$

Now, suppose a gene of antibody B^i to be

$$B^i = [4 \quad 3 \quad 1 \quad 5 \quad 0 \quad 2] \qquad (10)$$

Put $Y = A = [y(0), y(0), \cdots, y(5)]$ and the coding of B^i is as follows;

① $B_0^i[4 \quad 3 \quad 1 \quad 5 \quad 0 \quad 2], y(0)$
  $= 4(B_0^i) \rightarrow 4(K)$,
② $B_1^i[4 \quad 3 \quad 1 \quad 5 \quad 2], y(1)$
  $= 2(B_0^i) \rightarrow 2(K)$,
③ $B_2^i[4 \quad 3 \quad 5 \quad 2], y(2)$
  $= 3(B_1^i) \rightarrow 3(K)$,
④ $B_3^i[4 \quad 3 \quad 5], y(3)$
  $= 1(B_1^i) \rightarrow 1(K)$,
⑤ $B_4^i[4 \quad 5], y(4)$
  $= 0(B_4^i) \rightarrow 0(K)$,
⑥ $B_5^i[5], y(5)$
  $= 0(B_4^i) \rightarrow 0(K)$,

Therefor, K^i is

$$K^i = [4 \quad 2 \quad 3 \quad 1 \quad 0 \quad 0] \qquad (11)$$

Now, the non- singular matrix of G1 is Q,

$$(1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0) \qquad (12)$$

cut set matrix of A for $_A C_3$ is

$$(1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0) \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$= (0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1) \,(\text{mod } 2) \qquad (13)$$

⟨1⟩ cut segments $_A C_3$ is

$$_A S_3^{\langle 1 \rangle} = \{ A1, A4, A6, A7 \} \qquad (14)$$

⟨2⟩ turn the elements of $[0,1]$ for Eq.(13) is

$$(1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0) \qquad (15)$$

Referencing Eq.(15) and (G1) the matrix of inner matrix of non-cut set segments is

$$(0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0)$$

Then, the inner part of non-cut set segments $_A S_3^{\langle 2 \rangle}$ is

$$_A S_3^{\langle 2 \rangle} = \{ A2, A3 \} \qquad (16)$$

⟨3⟩ outer matrix is

$$(1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0)$$

Then, the inner part of non-cut set segments $_A S_3^{\langle 3 \rangle}$ is

$$_A S_3^{\langle 3 \rangle} = \{ A0, A5 \} \qquad (17)$$

The calculation is the same for G2. Referencing Eq.(10), apexes of G2 corresponding to apexes (0), (1) and (2) are as follows;

(4) $(\rightarrow G1:(0))$, (3) $(\rightarrow G1:(1))$, (1) $(\rightarrow G1:(2))$

Non-singular matrix of G2 is

$$(0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0) \qquad (18)$$

Cut set matrix is

$$(0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0) \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$= (1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0) \,(\text{mod } 2) \qquad (19)$$

⟨1⟩ cut segments $_B S_3^{\langle 1 \rangle}$ is

$$_B S_3^{\langle 1 \rangle} = \{ B0, B1, B2, B3, B5, B6 \} \qquad (20)$$

⟨2⟩ turn the elements of $[0,1]$ for Eq.(19) is

$$(0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1) \qquad (21)$$

Referencing Eq.(21) and (G2) the matrix of inner matrix of non-cut set segments is

$$(0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1)$$

Then, the inner part of non-cut set segments $_B S_3^{\langle 2 \rangle}$ is

$$_B S_3^{\langle 2 \rangle} = \{B7\} \tag{22}$$

$\langle 3 \rangle$ outer matrix is

$$(\ 0 \ \ 0 \ \ 0 \ \ 0 \ \ 1 \ \ 0 \ \ 0 \ \ 0 \ )$$

Then, the inner part of non-cut set segments $_B S_3^{\langle 3 \rangle}$ is

$$_B S_3^{\langle 3 \rangle} = \{B4\} \tag{23}$$

The cut set of $_A C_3$ is Eq.(14). Referencing the correspondence of Eq.(8) and Eq.(10), their attached relations and connection matrix Eq.(13) and Eq.(19) (left side second term), we correspond the A's segments to the B's segment as follows;

$$
\begin{aligned}
A1 \ ((0),(5)) \ &\to \ ((4),(2)) \ = B3 \\
A4 \ ((2),(3)) \ &\to \ ((1),(5)) \ = B0 \\
A6 \ ((0),(4)) \ &\to \ ((0),(4)) \ = B5 \\
A7 \ ((2),(5)) \ &\to \ ((1),(2)) \ = NC
\end{aligned}
$$

here, NC means Non Connect. The segment correspond to A7 is NC, the count of evaluation $_A S_3^{\langle 1 \rangle} = -1$.

In the same manner, we check the terms $\langle 2 \rangle$ and $\langle 3 \rangle$. For non-cut segments from the point of G2 to G1, we obtain the following results;

$$
\begin{aligned}
_A S_3^{\langle 2 \rangle} &= -2 \\
_A S_3^{\langle 3 \rangle} &= -2 \\
_B S_3^{\langle 1 \rangle} &= -3 \\
_B S_3^{\langle 2 \rangle} &= -2 \\
_B S_3^{\langle 3 \rangle} &= -2
\end{aligned}
$$

C_3 becomes,

$$C_3 = \sum_{i=1}^{3} \left(_A S_3^{\langle i \rangle} + _B S_3^{\langle i \rangle} \right) = -12 \tag{24}$$

4 Experiment

It is shown some examples of experiment in Figure 3,5,6,8,9, and the results of producted antibodies to f are shown in Figure 4, Figure 7 and Figure 10. All case are matched.

5 Conclusive Remarks

We proposed a matching algorithm of stereo segment images by using IA and demonstrated its effectiveness.
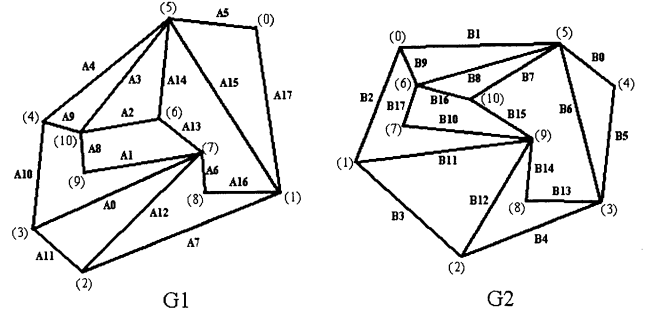
Figure 3: Ex.1
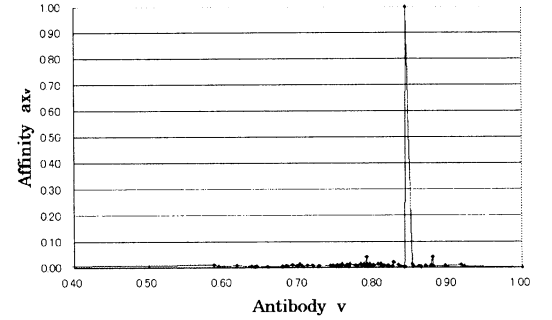
Figure 4: Producted antibodies to f (Figure 3)

References

[1] H.Mitsumoto, K.Okazaki, S.Tamura etc. : 3D Recognition Using Mirror Images Based on a Plane Symmetry Recovery Method, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.14, No.10, pp.921-927, 1992.

[2] K.Mori, M.Tsukiyama, T.Fukuda : Immune Algorithm with Searching Diversity and its Application to Resource Allocation Problem, T.IEE Japan, Vol.113-C, No.10, pp.872-878, 1993.

[3] J. Grefenstette, R. Gopal, B. Rosmaita and D. Van Gucht : Genetic Algorithms for the Traveling Salesman Problem, Proc. of 1st Int. Conf. on Genetic Algorithms and Their Applications, pp.160-168, 1985.

[4] D. Whitley, T. Starkweather and D'Ann Fuquay : Scheduling Problems and Traveling Salesman : The Genetic Edge Recombination Operator ; Proc. of 3rd Int. Conf. on Genetic Algorithms, pp.133-140, 1989.
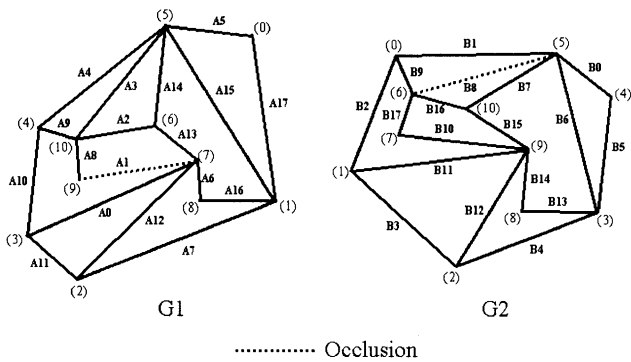
Figure 5: Ex.2
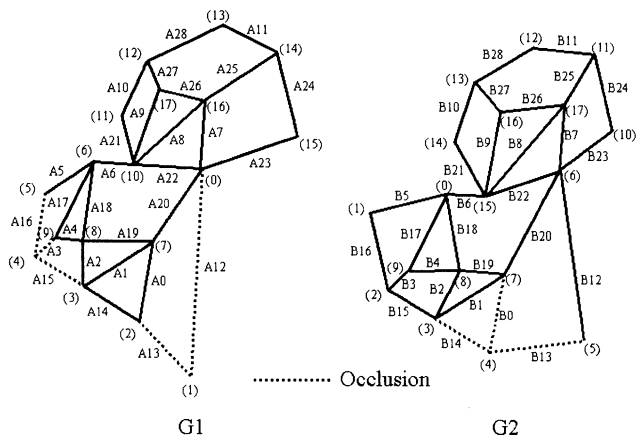
Figure 8: Ex.4
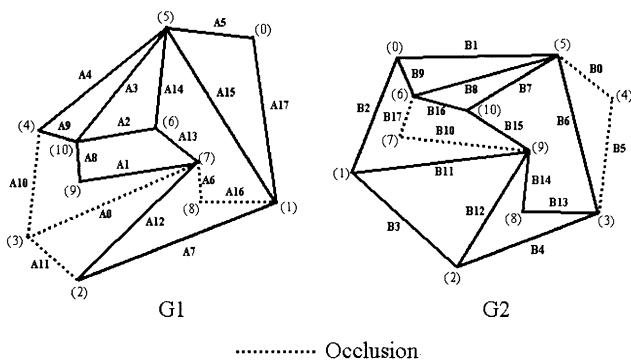
Figure 6: Ex.3
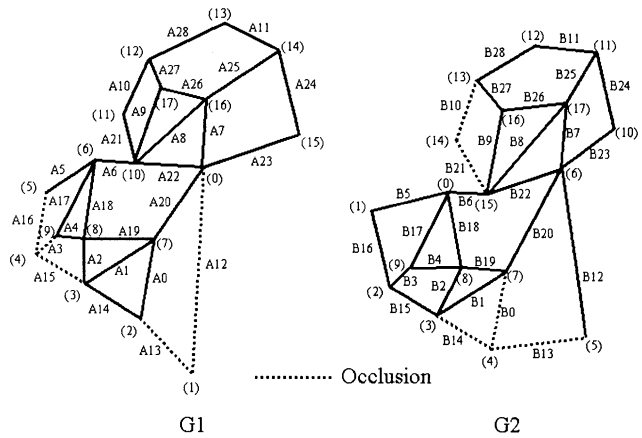
Figure 9: Ex.5
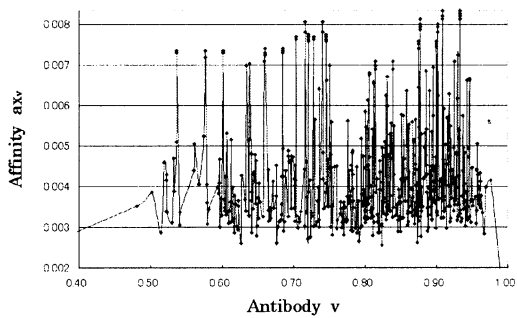
Figure 7: Producted antibodies to f (Figure 6)
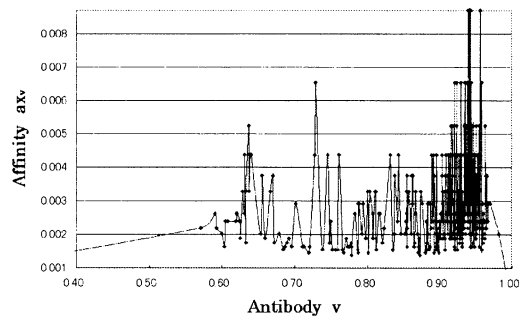
Figure 10: Producted antibodies to f (Figure 9)

Application of Genetic Algorithms for Minimizing the Consumption Energy of a Manipulator

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 robotic manipulators are used in the automated factory. In the point-to-point control, a manipulator's trajectory planning can be determined from various viewpoints. Many of these trajectory planning may give priority to operation efficiency, and it may not be taking the consumption energy. If the environmental problems are taken into consideration, we must decrease the consumption energy even of manipulators.

Since the dynamical equation of the multi-link manipulator is non-linear, it is difficult to solve the differential equation of two-point boundary value problem. In this case, the angle functions of the manipulator are approximated by the sequence of functions. The coefficients of the function space are searched by using the genetic algorithm so that the objective function can be optimized with satisfying the boundary conditions. This paper uses the Taylor series, Spline function and Fourier series for the approximating function. We propose a new searching method of genetic algorithm.

Keywords: Manipulator, Genetic Algorithm, Consumption Energy.

1   Introduction

There are many papers dealing with the trajectory planning of a manipulator. However, most of them are concerning the workability and the time optimal. Few of the paper attends to the saving energy of manipulators.[1] Considering environmental problems such as global warming of the earth, it is necessary to take into consideration for minimizing the consumption energy in a manipulator's trajectory control. In optimal control problem, if the system 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 and centrifugal force. Especially, when the joints of the manipulator have Coulomb friction, it is difficult to solve the differential equation of Lagrange's equation of motion.

The genetic algorithm is familiar with searching the solution in combinational optimization problems.[2] Moreover, it is also expected to obtain optimal solution for minimizing the objective function which is restricted by some complicated differential equations with strong non-linearity.

This paper describes the application of the genetic algorithm to the trajectory planning of the manipulator with non-linear friction and boundary conditions. Moreover, the paper proposes the technique not to lapse into the local optimal solution and the method to obtain a global optimal solution in a short time.

2   Dymamic Equation and Consumed Energy of 2-link Maniipulator

Figure 1 shows a horizontally articulated manipulator with two-joints which is considered in this paper. The angle and angular velocity vectors are function of time t as shown by $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t)]^T$ and $\boldsymbol{\omega}(t) = [\omega_1(t), \omega_2(t)]^T$, respectively. Dynamic equation of a manipulator is represented by

$$\boldsymbol{H}(\theta)\dot{\boldsymbol{\omega}} + \boldsymbol{C}(\theta, \omega) + \boldsymbol{F_r} = \boldsymbol{K}i(t) \tag{1}$$

where $\boldsymbol{H}(\theta)$ is the inertia matrix shown in Eq.(2), $\boldsymbol{C}(\theta, \omega)$ is the centrifugal and Coriolis's force vector shown in Eq.(3). Coulomb friction force $\boldsymbol{F_r}$ is made into a constant.

$$\boldsymbol{H}(\theta) = \left[\begin{array}{c} h_{11} + m_2 l_1 l_2 \cos\theta_2 \\ h_{21} + (m_2 l_1 l_2 \cos\theta_2)/2 \end{array} \right.$$
$$\left. \begin{array}{c} h_{12} + (m_2 l_1 l_2 \cos\theta_2)/2 \\ h_{22} \end{array} \right] \tag{2}$$

$$\boldsymbol{C}(\theta, \omega) = m_2 l_1 l_2 \sin\theta_2 \left[\begin{array}{c} -(\omega_1\omega_2 + \omega_2^2/2) \\ \omega_1^2/2 \end{array} \right] \tag{3}$$
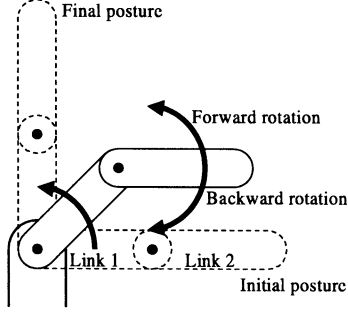
Figure 1: 2-Link Manipulator

where $h_{11}, h_{12} = h_{21}, h_{22}$ are constant depend on the moment of inertias about the center of gravity, m is mass, l is the length of links. The manipulator is driven by torque vector $\boldsymbol{K}\boldsymbol{i}(t)$. The vector $\boldsymbol{i}(t)$ is the current of motors rotating the joints. Consumption energy includes the Joule's heat loss due to the resistance $\boldsymbol{R}$ of the motor and mechanical loss due to friction. If the friction of the 2-link manipulator is ignored, the consumption energy of the manipulator moving in a time interval $[t_0, t_f]$ is given

$$J = \int_{t_0}^{t_f} \boldsymbol{i}^T(t) \boldsymbol{R}\, \boldsymbol{i}(t)\, dt \tag{4}$$

The manipulator is assumed to be moved from initial posture to final posture as shown in Fig. 1 . These postures give the following boundary conditions.

$$\theta(t_0) = \theta_0 \qquad \theta(t_f) = \theta_f \tag{5}$$
$$\omega(t_0) = 0 \qquad \omega(t_f) = 0 \tag{6}$$

Since $\theta(t)$ and $\omega(t)$ are continuous functions of time t due to the characteristic of a manipulator, they can be approximated by various function systems. In order to minimize the energy of Eq.(4), we propose the method of determining the coefficients of these approximated functions by using the genetic algorithm.

3 Optimization by Genetic Algorithm

3.1 Approximation Function

The angle and the angular velocity are continuous functions which determine the optimal trajectory. However, the genetic algorithm is the method of searching for the finite values in discrete solution space. The angle and angular velocity functions must be represented finite parameters which correspond to the genes in the genetic algorithm. This paper proposes the method of obtaining these parameters. The

differentiation of the angle function is the angular velocity, and that differentiation is the angular acceleration. Therefore, the approximation function has to be differentiable in second order. Moreover, it can be necessary to set the boundary conditions. Then, three approximation functions are considered as follows.

3.1.1 Taylor Series

The angle function is approximated by the Taylor series shown in Eq.(7). The angular velocity function is given by differentiation of Eq.(7) as Eq.(8).

$$\theta(t) = \sum_{i=0}^{M-1} a_i t^i \tag{7}$$

$$\omega(t) = \sum_{i=1}^{M-1} i a_i t^{i-1} \tag{8}$$

where M is the number of coefficients. These coefficients of the higher order are computed by the genetic algorithm search. This optimization problem has four boundary conditions of Eq.(5)(6). Therefore, coefficients of the lower order a_i ($i = 0, \cdots, 3$) are determined from these boundary conditions.

3.1.2 Spline Function

The method of using Spline function searches the positions of sample points. In order to take boundary conditions of the start and the final angles into consideration, the angles are given as edge conditions of the Spline function. Similarly, in order to take boundary conditions of the start and the final angular velocities into consideration, the angular velocities are given as edge conditions of the gradient of the Spline function. Therefore, the number of unknown design variables decides the number of sample points which do not contain edge points of the Spline function.

3.1.3 Fourier Series

An angle function can also be expressed by the Fourier series as

$$\theta(t) = A_0 + \sum_{n=1}^{M} \left(a_n \sin n\omega t + b_n \cos n\omega t \right). \tag{9}$$

The angular velocity function is written by

$$\omega(t) = \sum_{n=1}^{M} n\omega \left(a_n \cos n\omega t - b_n \sin n\omega t \right). \tag{10}$$

The number of design variables is $2M + 1$ in Eq.(9). The $2M + 1$ unknown design variables can be decreased
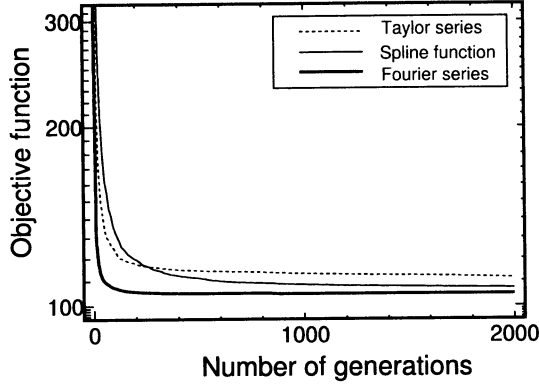
Figure 2: Transition of the objective function for the number of generations.

by 4 from 4 boundary conditions of Eq.(5)(6). In the Fourier series by contrast with the Taylor series, the lower order coefficients are computed by the genetic algorithm and the higher ones are determined from boundary conditions.

3.2 Transition of the Objective Function

The objective function is consumption energy of Eq.(4) to be minimized. The consumption energy were calculated 50 times for each method of Taylor series, Spline function and Fourier series. The transition of the averaged objective function are shown in Fig.2. In the first stage of the generation, the evolution of genes is advanced dramatically. This phenomena are the characteristic of the genetic algorithm. On the contrary, the evolution has declined in the last stage of the generation. For converged energy at 2000 generation, Taylor series is 111.5[J], Spline function is 107.2[J] and Fourier series is 104.8[J]. Most of the results have remarkable errors to the true solution of 102.3[J]. Especially, when Taylor series is used for the approximation function, it may lapse into the local optimal solution. They may exceed the consumption energy of over 130[J].

4 Proposed Method

In order to search for the minimum value of the consumption energy, the following method may be considered.

- Increment of the design variables.

- Using continuous solution space.

If the number of design variables is increased, an obtained solution may be approximated to the true solution more precisely. However, it makes increase the calculation time for obtaining the solution. It is presupposed that the number of the local optimal solution increases. If the discrete solution space becomes continuous, the approximation function is considered to become more accuracy. However, the unknown design variables is increased, it is difficult to express the optimal solution by the approximation function.

In order to examine the effect of increasing the number of the design variable, they are changed from 3 to 5 or 7. There are many local optimal solutions when the number of the design parameter is increased. And it was not able to escape from there to the global optimum. The method of using Fourier series can obtain the good solution as compared to the others. However, it lapsed into the local optimal solution in many cases. So, we propose the new method shown below.

- A few design variable is selected in the first half of the generation.

- The design variables are added in the second half of the generation.

The calculated solution in the first half of the generation is used to the initial genes of the second. In the second half, the genetic algorithm searches the optimal solution by using the increased genes. In order to examine these methods, the example is considered about three methods.

In the first half of a search generation, when the number of coefficients is 7, the method using Taylor series searches 3 unknown variables of a_i ($i = 4, 5, 6$) of Eq.(7). And, the other 4 variables of a_i ($i = 0, \cdots, 3$) are calculated by 4 boundary conditions. Then, when the number of design variables is change from 7 to 11 at the turn generation of the first half to the second half, the obtained solutions of the former are used as the initial gene's information of the latter. These obtained values of a_i ($i = 0, \cdots, 6$) are used for initial genes in the second half of a generation. And, the initial values of a_i ($i = 7, \cdots, 10$) are set to 0. The genetic algorithm searches for a_i ($i = 4, \cdots, 10$) of Eq.(7) in the second half of a generation. And, a_i ($i = 0, \cdots, 3$) are determined from boundary conditions.

In the Fourier series, searching is started with 7 unknown variables as same as the Taylor series. GA searches 3 unknown design variables of A_0, a_1, b_1 of Eq.(9) in the first half of a search generation. And, the other 4 variables of a_i, b_i ($i = 2, 3$) are calculated by 4 boundary conditions. Then, the number of design variables is change from 7 to 11 at the turn generation.
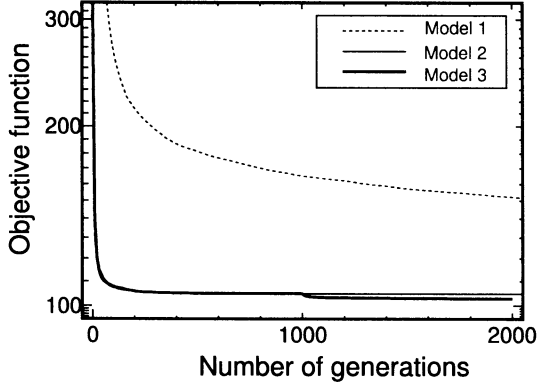
Figure 3: Transition of the objective function for the number of generations.

Figure 4: Angle functions

Table 1: Error distribution of solutions.

Error rate	Model 1	Model 2	Model 3
1[%]	85	0	50
2[%]	1	30	0
5[%]	8	20	0
10[%]	6	0	0
Other	27	0	0

The genetic algorithm searches the unknown values of A_0, a_i, b_i $(i = 1, 2, 3)$ of a Eq.(9) in the second half of a generation. And, a_i, b_i $(i = 4, 5)$ is calculated from boundary conditions.

The search using Spline function cannot be used the genetic information to the second half of a generation, it does not discussion in this paper.

5 Obtained Results

The methods of using Taylor series and Spline function could not find the good solutions, even if the number of coefficients is increased at a half of generations. So, we show the result only of using Fourier series.

The results of the transition of the objective function are shown in Fig.3. Three curves are the average of transitions in performed trial 50 times. A dotted line in Fig.3 shows the result of searching for 7 unknown design variables (Model 1). It is the way of evolution even at 2000 generation, and it has not obtained the good solution. A thin solid line shows the result of searching for 3 unknown design variables (Model 2). The obtained solution is not so bad. A thick solid line is result by using the proposed method. This method changes the number of genes from 3 to 7 at 1000 generation. The effect of the proposed method can be seen after changing the number of genes. The distribution of the error rate of the final solution is shown in Table 1. These numbers are calculated solution of consumption energy on the final generation (Model 1:10000, Model 2 & 3:2000). Even if Model 1 searches to the 10000 generation, the good solutions within 1 [%] were only 8 times. The obtained solutions of Model 2 were not bad, however, they are not at all within 1 [%]. In Model 3, all solutions of 50 times are

attained into within 1 [%] . Finally, we show the result of the angle function shown in Fig. 4. The obtained trajectory function satisfies the boundary conditions. The proposed method is useful for minimization of the consumption energy.

6 Conclusions

We described the method of the genetic algorithm of using approximation function. We showed the new method for improving accuracy of the solution and shortening the computation time. The proposed method is able to obtain the good solution for two points boundary value problem.

References

[1] Teruyuki Izumi, "Minimization of Energy Consumption for a Manipulator with Nonlinear Friction in PTP Motion ", *Journal of Robotics Society of Japan*, Vol. 13, No. 8, pp. 1179- 1185, 1995.

[2] O. A. Mohammed and G. F. Uler, "A Hybrid Technique for the Optimal Design of Electromagnetic Devices Using Direct Search and Genetic Algorithms ", *IEEE Trans. on Magnetics*, Vol. 33, No. 2, pp. 1931-1934, 1997.

The Distributed effect of the Real-coded GA

Masanori SUGISAKA and Toshiro KIYOMATSU

Department of Electric and Electronics Engineering, Faculty of Engineering, Oita University
700 Dannoharu, Oita 870-1192, JAPAN
Tel:+81-97-554-7841 Fax:+81-97-554-7818
E-mail: msugi@cc.oita-u.ac.jp, kiyomatu@cc.oita-u.ac.jp

Abstract

In this research, we studied about the distributed effect of the real-coded Genetic Algorithm (DRGA). With a general GA, it uses binary coding for coding process. Real-Coded GA (RCGA) does not need a coding process. Because the phenotype and the genotype are the same type for RCGA. And the crossover is different to bit type. Therefore, RCGA can search large scale. And then Distributed Genetic Algorithm (DGA) divides population into more than some sub-population and it does genetic among sub-population. Then it does the replacement (migration) of individual among sub-population every constant period (migration interval). In DGA, the division of the population maintains the variety of the solution and various schemata grow every sub-population. The better solution is generated by crossover among those schemata with migration. When comparing with single population GA, it is reported a high quality solution is gotten. This time, it proposes Environmental distribution type and elite sub-population generation type for Real-coded GA. Then we verify these techniques and are comparing a performance.

Key-word: Distributed genetic algorithm, Real-Coded GA, elite sub-population.

1. Introduction

The Genetic Algorithm (GA) is the probable optimal algorithm imitating evolution of a living thing. The technique can be adapted for the complicated continuation and a dispersed problem, which was difficult to solve by the conventional optimization technique. Furthermore, there is the strong point in which practical use is comparatively easy. However, since GA needs huge repeated calculation, its calculation cost is high.

Then, Distributed GA (DGA) that can calculate a solution in the amount of calculation fewer than the conventional GA according to the distributed effect attracts attention.

Regarding DGA, the research about various crossover and migration has been reported until now. However, we have still desired the improvement of efficiency about DGA. In addition verification is not made about the effectiveness of DGA in RCGA Therefore, we investigate the generation of sub-population, the interval and the time of migration in order to improve the efficiency RCGA.

2. Genetic Algorithm

2.1 Real-coded GA

In Real-coded GA, a chromosome of genotype is used real number. Traditionally, there is a bit string GA using a binary or gray expression as coding. As compared with that, coding of RCGA is unnecessary, and since it does not change, it is not dependent on resolution. And RCGA uses a real number vector directly. In the search, continuity is taken into consideration, there is also report that a solution search performance is obtained

to a continuous function optimization problem.

2.2 Crossover

Recently, the various crossover techniques are proposed to RCGA. It introduces to below about the technique, BLX-α and UNDX which we used this time.

(1) BLX-α

BLX-α generates a child at random according to a uniform variable from the section where only αI extended the section I of each variable parents' real number vector to both sides. In fact, each surrounding neighborhood of parents performs a random search in the domain of a super-rectangular parallelepiped parallel to an axis. BLX-α is shown in Fig.1.

$$c_{1i}, c_{2i} = u(\min(p_{1i}, p_{2i}) - \alpha I_i,$$
$$\max(p_{1i}, p_{2i}) + \alpha I_i) \qquad (1)$$
$$I_i = |p_{1i} - p_{2i}|$$

Where $P_1 = (p_{1i}, ..., p_{1n})$ and $P_2 = (p_{2i}, ..., p_{2n})$ are parents, $C_1 = (c_{1i}, ..., c_{1n})$ and $C_2 = (c_{2i}, ..., c_{2n})$ are children, $u(x,y)$ is the uniform random number of the section.
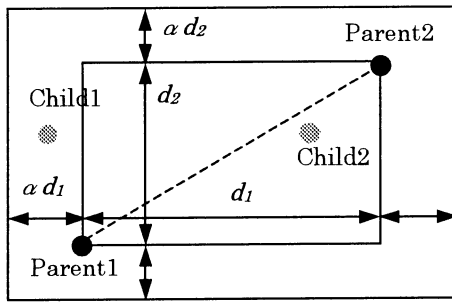
Fig.1 BLX-α

(2) UNDX

UNDX (Unimodal Normal Distribution Crossover) is shown in Fig.2. A child is generated according to the normal distribution it is decided by parents and the 3rd parents that will be the straight line top, which connects parents, and it's near. A child is generated not around an axis of coordinates, but around the axis to which parents are connected according to a normal distribution. Therefore, The probability to generate the distant child from axis connecting parents is low.

$$c_1 = m + z_1 e_1 + \sum_{k=2}^{n} z_k e_1$$

$$c_2 = m - z_1 e_1 - \sum_{k=2}^{n} z_k e_1$$

$$m = (P_1 + P_2)/2$$

$$z_1 \sim N(0, \sigma_1^2), \quad z_2 \sim N(0, \sigma_2^2), \quad (k = 2,...,n) \ (2)$$

$$\sigma_1 = \alpha d_1, \quad \sigma_2 = \beta d_2 / \sqrt{n}$$

$$e_1 = (P_2 - P_1)/|P_2 - P_1|$$

$$e_i \perp e_j, \quad (i \neq j) \ (i, j = 1,...,n)$$

Where P_1 and P_2 are parents, C_1 and C_2 are children, and m is the middle point of parents. The d_1 is the distance between parents P_1, P_2 and d_2 is the distance between 3rd parent P_3 (random select) and the line of parents. The z_1 is normal distributed random number of average 0, distribution σ_1^2. Tthe z_2 is normal distributed random number of average 0, distribution σ_2^2. The parameter α and β are constants value given by the user. The e_i is verticality to e_1 and linear independency unit vector.
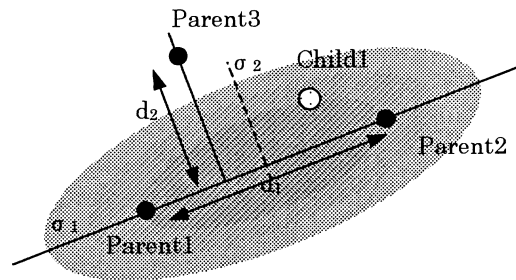
Fig.2 UNDX

2.3 Distributed GA

The population is divided into some sub-population in DGA, and genetic operation is performed independently for every sub-population. Also we exchange the individual information what is called migration among sub-population every constant interval. There is Migration interval and Migration rate as a unique parameter in DGA. Migration interval is defined as the rate of generation interval to migrate. As for featuring DGA, the division of population maintains the diversity of a solution, and various schemas grow for every sub-population. And then, a better solution is generated by transfer of those schemas by migration. In case that we want to get the solution with high probability, DGA is better than single population GA.

3. Experimental method

3.1 Environmental distribution type

In the usual DGA, the same sub-population type is used to parallel. This means that parameters, such as the crossover rate and the mutation rate are constant value in each sub-population. In this case, only the migration is used for diversity maintenance of a solution. There is a tendency depending on an initial population.

And so, we thought that environmental distribution type GA. There is DGA from which the parameter differed for each sub-population. From different parameters, diversity is maintained more.

3.2 Elite sub-population generation type

Generally, DGA makes the migration from one sub-population to another. The elite individual migrates another sub-population, be generated and the solution is obtained. So, we thought of make an elitist sub-population. If the elitist sub-population collected elite individual is made, that can obtain the solution earlier. So, we make genetic operation to them and we think that a still better individual will be

generated. We think that promotion of evolution and quality improvement of solution is expected by experiment on this operation.

3.3 MGG

The MGG (Minimal Generation Gap) is the desirable model in maintenance of diversity. Parents are chosen from maintenance of collective diversity at random, and a change of generation is performed locally. This technique of MGG used this time is described below.
1) select two parents for crossover in random from population 2) create children 3) evaluate each individual 4) select one elite individual and one individual randomly from parents and children 5) add selected two individuals to population
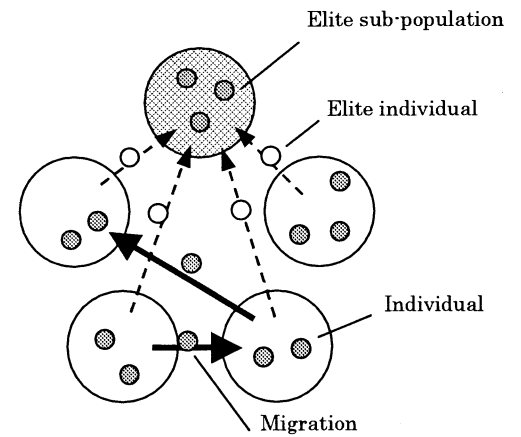
Fig. 3 Elite sub-population generation type

4. Simulation

In this time, it is comparing a performance in proposed DRGA and conventional GA. It uses a standard function in GA that Dejong proposed as the object problem (F1~F3). F1 is Sphere function, F2 is Rosenbrock function and F3 is Rastrign function. F1 is unimodal function, and does not have a dependency between parameters. F2 is unimodal function and has a dependency between parameters. F3 is multimodal function and does not have a dependency between parameters. And then, we set the design parameter of each function to 10.

$$F1: \quad -\sum_{i=1}^{n} x_i^2 \qquad (-5.12 \le x < 5.12)$$

$$F2: \quad \sum_{i=1}^{n-1} \left\{ 100\left(x_i - x_{i+1}^2\right)^2 + \left(1 - x_i^2\right) \right\}$$

$$(-2.048 \le x < 2.048)$$

$$F3: \quad 10n + \sum_{i=1}^{n} \left\{ x_i^2 - 10\cos(2\pi x_i) \right\}$$

$$(-5.12 \le x < 5.12)$$

5. Experiments and Results

In the experiment, we use individuals 300, crossover rate 1.0 and mutation rate 0.1. And we use migration rate 0.5, migration interval 5. The experiment results are average of 10 trials.

As a result of the experiment, the following graph was gotten (Fig.4, 5).
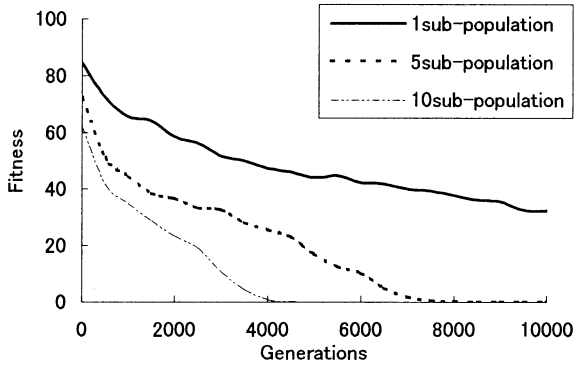
Fig.4 The effect of distribution

The effect by the distribution of Real-coded GA in F3 is shown in Fig. 4. In this experiment, we used BLX-α as the crossover method. The value of α is 0.5.

Compared one sub-population, the good performance was shown. By distributing, this is considered because the number of individuals was reduced and it searched efficiently.

Moreover DRGA affects the number of sub-population.

The result of the influence by environmental distribution is shown in Fig. 5. In this experiment, we used BLX-α as the crossover method and five sub-populations.

As environment distribution, each sub-population parameter α and mutation rate
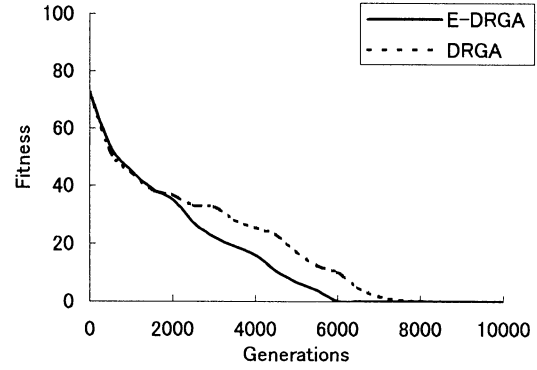
Fig.5 Environmental distribution type

we used. In this time, we used {0.3, 0.5, 0.7} as value of α and {0.001, 0.01, 0.1} as value of mutation rate for sub-populations.

As compared with usual DGA, it turns out that environmental distributed type is converged early. Because of the search methods differ by the each sub-population, the convergence comes out early.

6. Conclusions

In Real-coded GA, the distributed effect was verified. We obtained the result good for DRGA as compared with single RCGA. But DGA is dependent on parameters, we need to study the parameters in DRGA.

References

[1] H.Kitano, "Genetic Algorithm 4", Sangyotosho, 2000.

[2] D.Whitley, K.Matis, S.Rana, and J.Dzubera, "Building better test function", International sConference on Genetic Algorithm, 1995.

[3] M.Sugisaka and K.Mori, "Improvement of the efficiency of Distributed GA", Proc. of AROB 7th, 2002.

[4] I.Ono, M.Yamamura, and H.Kita, "A Real-Coded Genetic Algorithms and Their Applications", Proc. of JSAI vol.15 (2), 2000.

[5] M.Miki, T.Hiroyasu, J.Yoshida and I.Omukai, "Optimal Crossover Schemes for Parallel Distributed Genetic Algorithms", MPS symposium, 2000

The improvement of the diversity and the searching ability in GA

◯ Minoru ITO and Masanori SUGISAKA

Faculty of Engineering, Oita university

700, Oaza Dannoharu, Oita, 870-1192, JAPAN

minoru@cc.oita-u.ac.jp , msugi@cc.oita-u.ac.jp

Abstract

It is one of the important problems to maintain the diversity in genetic algorithm. Generally, when the diversity is not fully maintained, it is difficult to find good solution, and it is easy to stagnation the premature convergence. In this paper, we propose an Elite Correlation Selection (ECS) operator as a new selection operator to improve the genotype diversity in genetic search process. In the proposed method, a candidate for selection are evaluated the genotype similarity among individuals, and the individual is selected based on the above new evaluated values. As a measure of the similarity, we use the hamming distance of the genotype. The proposed method also is the improvement of the selection operator of the Minimal Generation Gap (MGG) model. The performance of the proposed method is evaluated with the standard test functions. The performance of the proposed method are compared with the MGG model, the Steady State Genetic Algorithm (SSGA) and the Simple Genetic Algorithm (SGA). The simulation results show that the proposed method is very effective.

Keywords: Genetic Algorithm, Selection Operator, Generation Alternation Model

1 Introduction

A Genetic Algorithm emulates biological evolutionary theories to solve optimization problems. A GA is the stochastic search and optimization algorithm. A GA searches the solution based on the biologically inspired operator (crossover, selection and mutation operator) in multiple searching points[1].

In this GA, the SGA is one of well-known models[2]. In the SGA, we select individuals using the roulette selection based on the fitness values. In addition, we replace all individuals in population with the generated individuals. In the above selection operator, there is the following two problems. One is the problem of the "premature convergence" in the first stage of search process, the other is the problem of the "evolutionary stagnation" in the middle and last stage of search process[3]. From the above problems, we need to consider the new selection operator which makes the various individuals survive in the population.

In order to maintain the diversity of population, many improvements are proposed. As a major methods, the MGG model proposed by Satoh et al.[4] and the SSGA proposed by Syswerda[7]. Generally, the GA has two types of the selection operator[5]. One is the "Selection for Reproduction", the other is the "Selection for Survival"; the former is for crossover and the latter is for individuals which survive to the next generation. In the MGG model, we select two individuals for the selection for reproduction. For the selection for survival, we select two individuals by elite selection and roulette selection based on the fitness values. From these selection operators, the MGG model prevent the premature convergence in the first stage of search and the evolutionary stagnation in the middle and last stage of search. However, this selection operator uses only fitness values in the MGG model. We apply the similarity among individuals to the selection operator, and we can expect the high genotype diversity.

In this paper, we propose the ECS operator as a new selection operator. The proposed method is an extension of the selection for survival in the original MGG model. In this method, we select one individual by elite selection, and we evaluate the similarity between elite individual and the other individuals. We select other one individual by roulette selection based on the similarity. In this paper, as a measure of the similarity, we use the hamming distance of the genotype. The hamming distance is the count of bits different in the two individuals. In this paper, we evaluate the proposed method using standard test functions, and we compare the proposed method with both the original MGG model and the SGA. We show the effectiveness of the proposed method in the third section.

2 Proposed method

In this section, we propose the ECS operator as a new selection operator to maintain the genotype diversity.

In the first subsection, we review the MGG model as the basic model of the proposed method. In the second subsection, we propose the ECS operator in order to improve the genotype diversity.

2.1 Minimal Generation Gap : MGG

The MGG model proposed by Satoh et al. is an effective one of models. In this MGG model, we randomly se-

Fig 1: Minimal Generation Gap.

Fig 2: Elite Correlation Selection.
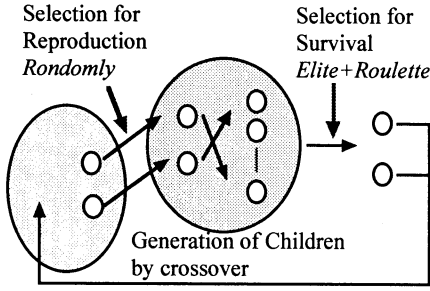
lect parents from population, and we generate children by crossover. From the parents and their children, we select the elite individual and a random one using the roulette selection based on **the fitness values**. The original parents are replaced by the selected children. In this model, the original parents are two individuals, and replacing individuals are also two. It should also be noted that this model only uses fitness values of each individuals. Fig 1 shows the outline of the MGG model. In this model, the selection operators are summarized as the following.

- Selection for Reproduction : Randomly.

- Selection for Survival : Elite + Roulette (fitness values).

2.2 Elite Correlation Selection : ECS

In this subsection, in order to improve the genotype diversity, we propose the ECS operator as a new selection operator for survival in the original MGG model. Fig 2 shows the outline of the proposed method. In this figure, the number in brackets corresponds to the serial number of the following procedure.

In the proposed method, like the MGG model, we randomly select parents, and we generate children by crossover. As a crossover operator, we use the 2-point crossover. From the parents and their children, we select the elite individual and a random one using the roulette selection based on **the genotype similarity**. The original parents are replaced by them. Then we leave the elite individual for solving problem, and leave a random low similarity individual for maintaining the genotype diversity of population.

The procedure of the ECS is as following:

1. Initialize and evaluate P.

2. Randomly select C from P.

3. Create children C' from C.

4. Select elite E from C'.
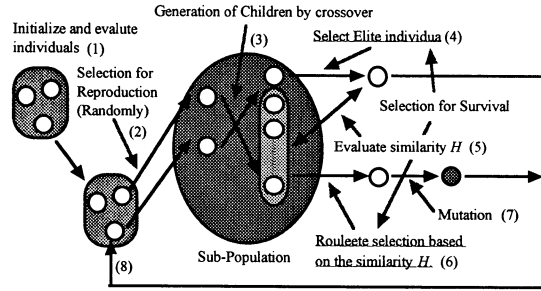
5. Evaluate similarity H.

$$H_i = |h_i - h_e| \qquad (1)$$

H is the similarity, h_i is the hamming distance in each individuals, H_e is the hamming distance of elite individual.

6. Select other one R form C'.

7. Mutation R.

8. Replace.

9. repeat from 2 to 8 until it satisfy the terminal condition.

For this method, we can expect the proposed method to maintain the high genotype diversity and contribute to the high searching ability. In the proposed method, we use the hamming distance as a measure of the similarity H.

In the proposed method, the selection operators are summarized as the following.

- Selection for Reproduction : Randomly.

- Selection for Survival : Elite + Roulette (similarity).

3 Simulation and Results

In order to confirm the effectiveness of the proposed method, we compare the ECS with the original MGG model, the SSGA and the SGA using several standard test functions.

In the first subsection, we show the test functions. In the second subsection, we show the experimental setting. In the third subsection, we show the simulation results and discuss the simulation results.

3.1 Test functions

In this paper, the performance of the proposed method is examined with standard test functions[6]. The optimization problems used here are the minimization of

Table 1: Test functions and its domains.

$F_1(x_i) = \sum_{i=1}^{n} x_i^2$
$-5.12 \leq x_i < 5.12$

$F_2(x_i) = \sum_{i=1}^{n} 100(x_1^2 - x_2)^2 + (1 - x_1)^2$
$-2.048 \leq x_i < 2.048$

$F_3(x_i) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right)$
$-5.12 \leq x_i < 5.12$

$F_4(x_i) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \left(\cos\left(\frac{x_i}{\sqrt{i}}\right)\right)$
$-512 \leq x_i < 512$

Table 2: The number of variables, gene length per variable, and total length in test functions.

Functions	#var.	#gene/#var.	#total length
F1	10	10	100
F2	5	12	60
F3	10	10	100
F4	10	20	200

Table 3: Experimental setting.

	ECS	MGG	SSGA	SGA
Crossover	2X	2X	2X	2X
Crossover Rate	1.0	1.0	1.0	0.8
Mutation Rate	0.05	0.05	0.05	0.05
Population Size	50/100	50/100	50/100	50/100
Max Generation	10000	10000	10000	10000
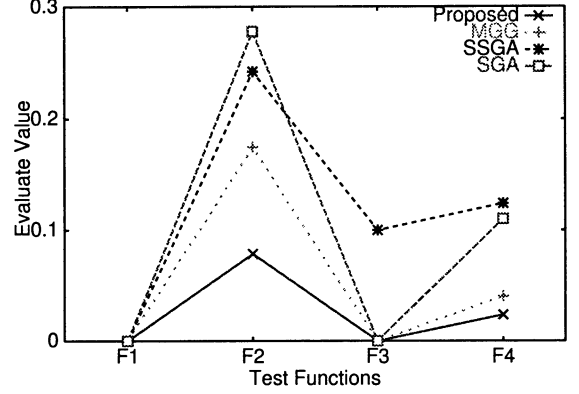
Fig 3: Function Values at last generation.

the Sphere function (F1), the Rosenbrock function (F2), the Rastrigin function (F3), the Griewank function (F4). Table 1 shows the test functions and its domains.

The Sphere function is the unimodal function and it has the independence on its variables. The Rosenbrock function is the unimodal function and it has the dependence on its variables. The Rastrigin function is the multimodal function and it has the independence on its variables. The Griewank function is the multimodal function and it has the dependence on its variables.

The unimodal functions have only one global optima. On the other hand, the multimodal functions have many local optima, but only one global optima.

Table 2 shows the number of variables (#var.), gene length per variables (#gene/#var.), and total gene length (#total length) of all test functions. The all variables in the chromosome are encoded by gray code.

3.2 Experimental setting

In this paper, we compare the performance of the proposed method with the original MGG model, the SSGA and the SGA. An experimental setting is shown in Table 3.

We use the 2-point crossover (2X) as a crossover operator in all methods, and the crossover rate are 0.8 and 1.0. In all methods, we use the mutation operator, and the mutation rate is 0.05. In the SGA, we use the roulette selection and elite selection as a selection operator. In the selection for SSGA, we replace the worst individuals with the new individuals. In the each methods, the population size is 50 and 100, and the

max generation is 10000 generations.

3.3 Simulation results

Fig 3 shows the best function values at the 10000 generations. In this figure, the y axis is the function values and the x axis is the test functions. These results are the average of 10 trials. We see from this figure, the proposed method approached the best function values.

Fig 4,6 show the histories of the best function values on the Sphere function and Griewank function, respectively. These results are the average of 10 trials. In these figures, the y axis is the function values and the x axis is the generations. The x axis also is log scale. In these figures, the convergence speed of the proposed method is slower than the conventional methods. This point, the selection pressure of the proposed method is lower than the conventional methods.

Fig 5,7 show the histories of the genotype diversity on the Sphere function and the Griewank function, respectively. These results are the average of 10 trials. The genotype diversity (GD) is given by

$$GD = \frac{1}{N} \sum_{i=1}^{N} (h_i - \bar{h})^2 \qquad (2)$$

where N is the population size, and h_i are hamming distance of each individuals, and $\bar{h}$ is the average of hamming distance of population. In these figures, the y axis is the genotype diversity and the x axis is the generations. The x axis also is log scale. In these figures,
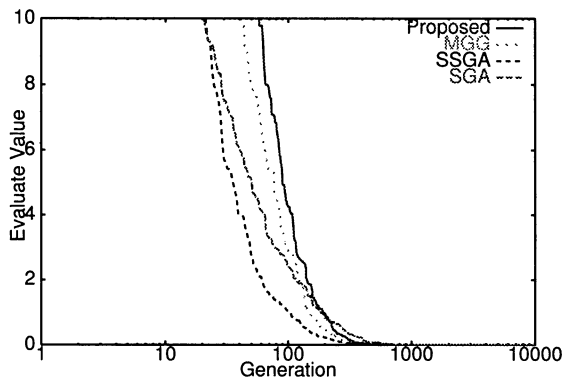
Fig 4: History of the best function value in F1.
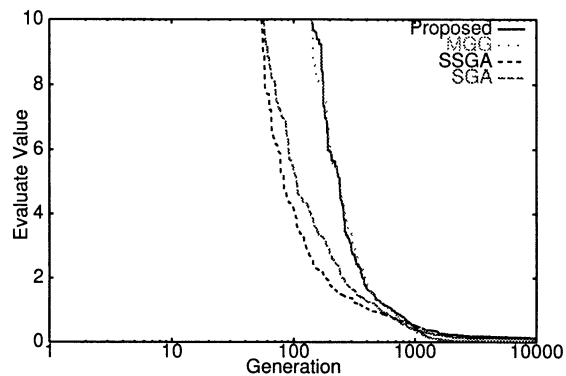
Fig 6: History of the best function value in F4.
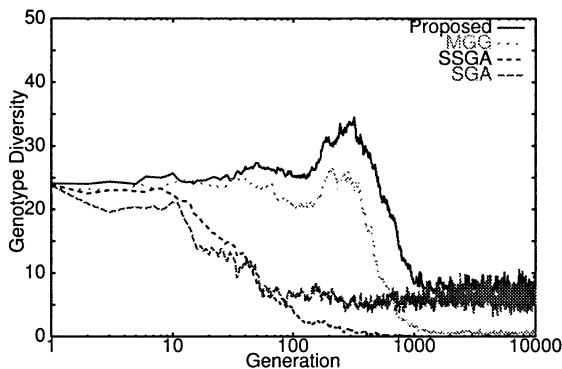
Fig 5: History of the genotype diversity in F1.
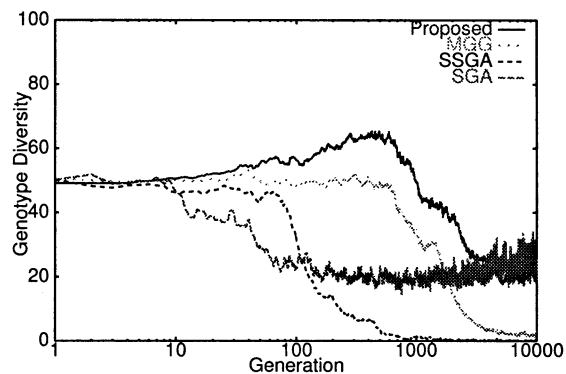
Fig 7: History of the genotype diversity in F4.

the proposed method maintain the genotype diversity in all search stage.

From these results, the proposed method can maintain the high genotype diversity. This high genotype diversity contribute the high searching ability.

4 Conclusion

In this paper, we proposed the ECS operator as a new selection operator for survival to improve the genotype diversity. We compared the performance of the proposed method with the conventional methods (the MGG model, the SSGA and the SGA). The proposed method showed higher performance than the conventional methods. The simulation results of all experiments suggested that the proposed method has a high diversity maintaining ability and the high searching ability.

References

[1] H.Kitano, "Genetic Algorithm," Sangyou-tosyo, 1993, in Japanese.

[2] D.E.Goldberg, "Genetic Algorithm in Search Optimization and Machine Learning," Addison-Wesley, 1989.

[3] D.Whitley, "The GENITOR Algorithm and Selection Pressure : Why Rank-Based Allocation of Reproductive Trials is Best," Proceedings of 3rd International Conference of Genetic Algorithm, pp. 116–121, 1989.

[4] H.Satoh, M.Yamamura, S.Kobayashi, "Minimal Generation Gap Model for GAs considering Both Exploration and Exploitation," proc. of IIZUKA'96, pp. 494–497, 1996.

[5] H.Satoh, I.Ono, S.Kobayashi, "A New Generation Alternation Model of Genetic Algorithms and its Assessment," Journal of the Japanese Society for Artificial Intelligence, Vol.12, No.5, pp. 734–744, 1997, in Japanese.

[6] D.Whitley, K.Mathias, S.Rana, and J.Dzubera, "Building better test function," International Conference on Genetic Algorithm, 1995.

[7] G.Syswerda, "Uniform crossover in genetic algorithms," Proceedings of the 3rd International Conference on Genetic Algorithms, pp. 2–9, 1989.

Virtual and Real Robots Through Interactive Web-Based Multi User 3D Virtual Environment

Maki K. Habib

School of Engineering and Science, Monash University Malaysia
Bandar Sunway, Petaling Jaya, Selangor, Malaysia
maki@ieee.org

Abstract

This paper presents a new concept that provides users access to remote virtual and real facilities (robots) through virtual environment. This system integrates a virtual laboratory to help in transferring theoretical knowledge into practice and enables geographically dispersed people to interact individually or in a group by immersing them through shared interactive 3D virtual environment. This virtual environment enables individual users manipulating data, manipulating objects, remote diagnostics and analyses, and simultaneously control, monitor, view and discuss results within a team as if they are all were interacting with each other in the same place. This concept has been implemented and a virtual robot that mimics the robot structure, control panel, along with operations and control features of a real robot called KUKA KR-6 has been integrated. Authorised users can access both physical and virtual robots. The physical robot has been integrated to facilitate telecontrol operations in real time through the interactive 3D virtual environment. Three modes of operations have been implemented, individual robot training through virtual robot, multi user mode, and group-based training by instructor. Also, The control of the real robot is facilitated by simulation and direct control while enabling monitoring during command execution.

Keywords:

Virtual Environment, Web Based, Internet, Interactive User Interface, Virtual Robot, Virtual Education, Virtual Learning, Telecooperaion.

1. Introduction

Information, expertise, facilities and other resources are scattered across the globe and may not be available where they are needed. The quality of communication technology and infrastructures of networking are forcing for definite move towards a rethinking of the physical structures and spaces needed for living, working and learning. As physical presence is no longer a prerequisite for physical influence there is a need to look for alternative forms of physical movement of people, equipment, records and information, etc. This approach is evolving to address the issues of universal sense of connectivity and global access to information, easy control and use of globally linked physical devices and processes, facilitating real time interaction and awareness along with supporting mobility and portability of people and enhances access to experts and other dispersed and valuable resources.

Telecooperation is an outstanding example for the power of enabling technologies as well as it is the latest appearing manifestation of those guiding visions and paradigms that have governed application development. Telecooperation stands for the fusion of computer science, telecommunication and multimedia facilities to carry out a cooperative process among organizations, individuals, systems or combination of them by having better access to resources and share timely information over geographically distributed locations that lead to make a well informed decisions.

The most suitable environment that realizes telecooperation among dispersed teams is the Internet. The Internet represents the suitable infrastructure technologies that facilitate virtual presence via electronic communications between entities, human populations and machines. The Internet provides low cost and widely available media rich interface that can make resources accessible to a broad range of users regardless of geographic location and the type of operation system they are using. Internet based systems with tools supporting human communication and interaction with real world, will inevitably lead to many useful applications in various sectors of the society, professionally and socially. Internet is a rich, interactive and engaging environment and it is a challenging way of providing training and education to assist learner in developing knowledge and skills through a well planned, well designed educational/ training environments.

Within the global and rapid development of educational needs "virtuality" indicates to the aspect of introducing a telematic network into the process and interaction of teaching and learning. The characteristic of virtuality encorporates a vision that no physical campus or training center as a separate organization will be needed but that learning place with its facilities and supporting organization may be created as a virtual network of elements and contributions of different organizations. The virtual campus refers to the environment that supports its research/educational and administrative activities that leap over the boundaries of time and space.

2. Web Based Virtual Robotics Education

A web based virtual robotic educational system has been developed. This system contains the duplicate of an actual industrial robot that is integrated within a 3D interactive virtual environment delivered through the Internet browser. The virtual robot that mimics robot's structure, control panel, along with operations and control features of a real robot KUKA KR-6. The developed system helps users to have real-like experiences of the most of functionalities of physical industrial robot system. Also, enables users to learn about robots individually and cooperatively and to enhance their skill beyond time and space constraints. Users can observe basic kinematics in robotics including robot configuration, joint movement, mathematical transformations and coordinates. The virtual robot can be accessed by unlimited number of clients (depend on the resources of the server) through 3D web-based environment that support multi-user access simultaneously. With this virtual model, an authorised user can interact with other users, move through the environment to see the robot from any angle, and go as close or as far away while interacting with the environment. The real industrial robot is integrated with the environment and an authorised user can establish a link between the virtual world and the physical robot and interact with it through integrative and flexible user interface. Technically, the concept and architecture of the developed web based virtual robotic system was implemented on the bases of VRML as a tool for virtual reality and 3D modelling. In addition, the External Authoring Interface (EAI) and Java programming have been used. A system with such features can support cost effective approach for training, program development, and safety during remote task execution. The physical robot has been integrated to facilitate telecontrol operations in real time through the interactive 3D virtual environment. Three modes of operations have been implemented, individual robot training through virtual robot, control of a real robot, and group-based training by instructor. The affect of time delay over the Internet has been studied for the purpose to overcome or minimise its effect. Fig. 1 shows the overall structure of the developed system

3. The Interactive 3D Virtual Environment

Merging virtuality and reality in unique and innovative ways takes new meanings and its implication is extensive. Users of the system are immersing themselves into a 3D based virtual environments, that create illusion of physical presence at the physical place where the robot and other equipment are available and supported by sensory devices that can develop a better understanding and appreciation of reality. The 3D virtual environment is easy re-configurable and it is divided into segments called zones. The functional capabilities of each zone are adaptable and it is not necessary to be the same in all zones. Users of the system can move between the zones

of the virtual environment seamlessly. The appearance and capabilities of graphically represented task environment as space, equipment and people are displayed and tracked when they are logged in to the system. The system can be described as a four-tier architecture, beginning with the device-level hardware (sensors and equipment to be controlled), the process server (which handles the direct communication between the central server and the hardware), the central server (which handles client administrative functions), and the clients themselves.
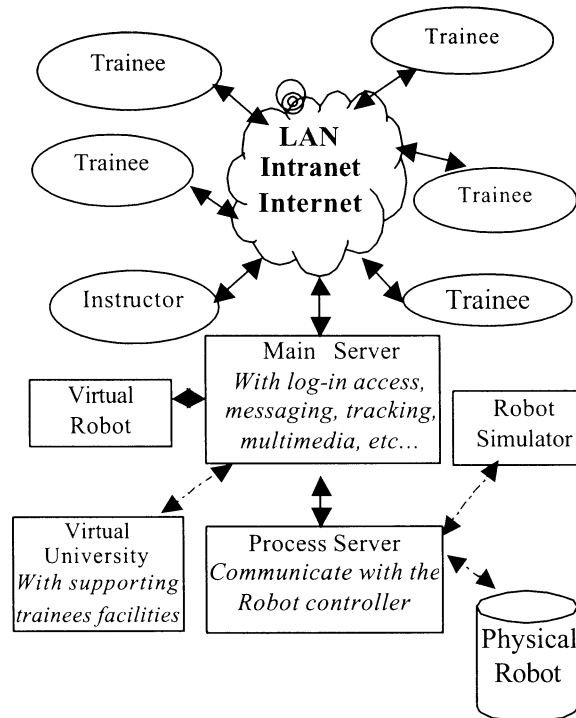
Figure 1: Overall System Architecture

The 3D virtual environment that include the virtual and real robots and enabling multi-users has the following structure:
1. A virtual environment that reflects a university campus.
2. Training center through virtual robots. This center includes training cells to facilitate training using virtual robot.
3. A virtual robot laboratory that integrate physical robots and has simulation capabilities.
4. The virtual robot and its virtual control panel that mimic the real one.
5. A high-end seamless integrative and interactive user interface that enable the interaction with the virtual robot, with the physical robot and with the users accessing the environment.
6. Physical camera and sensors to facilitate monitoring and feedback about the physical environment along with conferencing and awareness puposes.

4. The Client User Interface

Intuitive, coherent, integrated and multimedia based user interfaces represent a trade off between ease of use and the capacity for complex task. The designed client side of user interface maximizes quality information transfer, and the usability of wide range of users. The user interface allows the operator to navigate the virtual campus and to have seamless control over the virtual and physical facilities. The user interface represents the system in an abstracted hierarchy that details the structure of the system at different level of abstraction. The need to facilitate team members perception through proper information transfer and a better situational awareness, and to have variety of command generation tools that enable and speed up efficient and accurate motion command generation. Fig.2 show the hierarchical structure of the Client user interface.
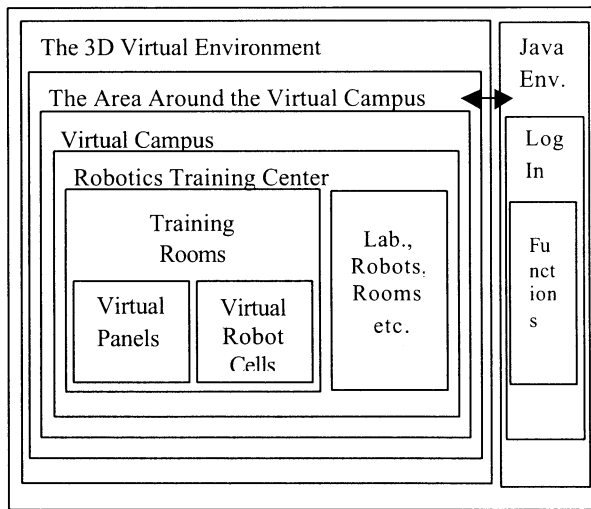
Figure 2. The hierarchical structure of the Client User Interface

5. The Server

The main server is acting as the center of the system. It basically acts as the provider of client administrative functions. All features of the system, whether multi-user, chat, or control, will have to pass through the central server. On a related note, the central server handles the task of first-level security or authentication. Access-Control is a feature integrated to the central server. Each command sent by a client is checked against his/her permission group and allowed or denied as defined by the designated permissions. The server also runs a basic logging service, which logs important events such as logins, logouts and access violations. To perform teleoperation with physical robots, the server must make a secure connection with the process server to enable data interchange.
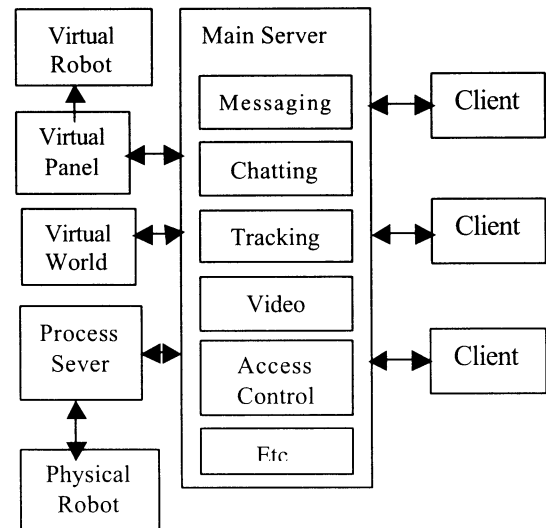
Figure 3: The Main Server

6. The Physical and Virtual Robot

The physical robot KUKA KR-6 has been simulated and integrated within the 3D virtual environment. Fig. 4 shows the laboratory environment with the interactive user interface that enables control and test through simulation and direct telecontrol of the physical robot.
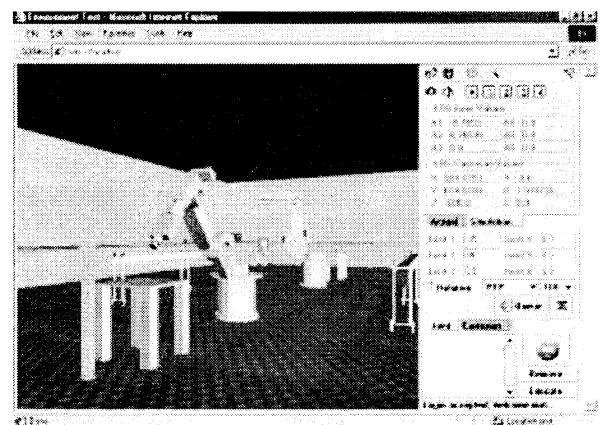
Figure 4. The user interface that enables interaction with the robot through simulation and direct telecontrol.

The potential applications of having a virtual robot are to use it for training and educational purposes, where students and trainees can experience programming and controlling a real-like robot. Another important application area is in testing and validation, whereby users may test their commands and programs within the simulation environment before applying it to the actual

robot. Through this potential errors can be detected and fixed, and risk can be minimized. Fig. 5 shows the virtual robot with its virtual control panel. The user can interact with the robot through the control panel by following the exact procedures that applied to the real robot.
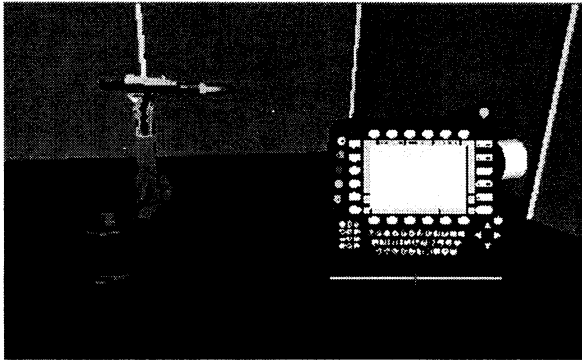
Figure 5. The virtual Robot with the Virtual Control Panel

7. Modes of Training

Three modes for training have been designed and integrated with the developed system. In each modes, the user will have different functional capabilities provided by the system that is dependent according to the requirements of that mode. Brief explanation about the three modes is shown below:

7.1 Individual Training Mode
In the Individual training mode, the user is given the right to learn how to control the virtual robot within a virtual training cell where a virtual control panel and a virtual robot are available. Upon entering this mode, the user will be able to navigate inside the environment of the virtual cell and move around the robot while the panel control will be always in the view of the user. In this mode, each trainee has the opportunity to control the virtual robot that he/she initiates. Currently there are 10 virtual training cells and users can access and use them simultaneously. Each virtual cell is dedicated to one user.
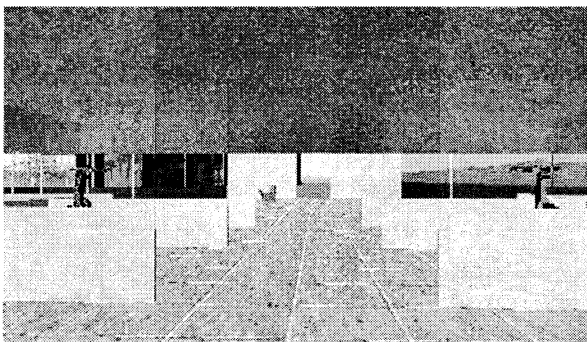
Figure 6. View of the Virtual Training Cells

7.2 Multi-user Training Mode
The multi-user mode has been implemented to enable group of users (more than one) to communicate, coordinate and cooperate with each other inside the shared virtual cell. The users can recognize each other through avatars with a name attached to it representing each user. All communication, messaging, conferencing, tracking and other functionalities integrated with the virtual environment are available for the users to use

7.3 Instructor Guided Training Mode
The reason for this mode is to provide a real time guide through an expert for the trainees to enable them to refer or to discuss with when they are face problems during their training. Also, the instructor may have a class with multiple users to demonstrating either a virtual robot programming or an implementation and execution of programs on the physical robot integrated with the system.

8. Conclusions

In this paper a new concept that provides users access to remote virtual and real facilities was presented though collaborative 3D virtual environment. The presented system enables geographically dispersed people to interact (communicate, cooperate and coordinate) individually or in a group by immersing them through shared interactive 3D virtual environment. The main purpose of the developed virtual robot system (but not limited to) is the education/training of robotics as well as web-technology and virtual reality. Current work is focusing on enhancing robot functionalities, seamless interaction capabilities, awareness capabilities among users and environment, and adding more features to the currently implemented training modes.

References

[1] D.W., Calkin, R.M., Parkin, R., Safaric, C.A., Czarnecki, "Visualisation, simulation and control of a robotic system using Internet technology" 5th International Workshop on Advanced Motion Control nAMC'98, pp. 399-404, 1998.

[2] E. Kruse, and F. Dai "Virtual Training Using Web and Interactive Visualisation", 1st International Conference on , pp. 517-522, 2000.

[3] E., Wagner, "Creating a Virtual University in a Traditional Environment", Proceedings of the 7th European Distance Education Network, Italy, 1998.

Virtual Walkway System with a New Gait Simulator

N. Shiozawa   M. Kishiba   M. Makikawa

Graduate school of science and engineering,

Ritsumeikan University

1-1-1 Noji-higashi, kusatu, Shiga, 525-8577, Japan

Abstract

In this study we have developed a new gait simulator to get more freedom of walking styles, that is, straight walking, direction change, going up and down stairs and so on. Moreover we have developed a Virtual Walkway System by combining this gait simulator with a HMD as a virtual display.

The gait simulator has two foot plates and the user stands on these plates. Simulation procedure for straight walking is as follows; 1) the right foot plate follows the right foot while right foot is in swing phase, and the left foot plate pulls back the left foot during stance phase, 2) all foot plates stop during stance phase of both feet, 3) the left foot plate follows the left foot during swing phase, and the right foot plate pulls back the right foot during stance phase. The user can freely stop, because all foot plates stop during stance phase of both feet.

The results show that the user can walk straight at 3 km/h and can go up and down stairs on our gait simulator.

1   Introduction

Virtual reality that allows man to experience variety things often needs man to walk in a virtual space. Therefore a gait simulator is necessary to realize the walking in the virtual space. Many attempts have been devoted to develop a gait simulator. Of course the basic gait simulator is a treadmill. However, this treadmill can simulate only straight walking. Darken and others developed Omni-directional Treadmill [1] and Iwata developed Torus Treadmill [2]. These systems have two perpendicular 2 DOF treadmills. User on these treadmills can walk straight and change direction. Hirose employed a handle to change direction [6]. Iwata and Fujii developed a sliding interface. In this system the user wears roller-skates or shoes with low-friction films and is rigidly attached to a frame by a belt set around the waist [3, 4]. Iwata and others developed Gait Master with two 6 DOF motion platforms [5]. Gait Master simulates omni-directional uneven surface.

In this study we have developed a new gait simulator to get more freedom of walking modes; straight walking, direction change, going up and down stairs and so on. Moreover we have developed Virtual Walkway System by combining this gait simulator with a HMD as a virtual display. Figure 1 shows the picture of our Virtual Walkway System.
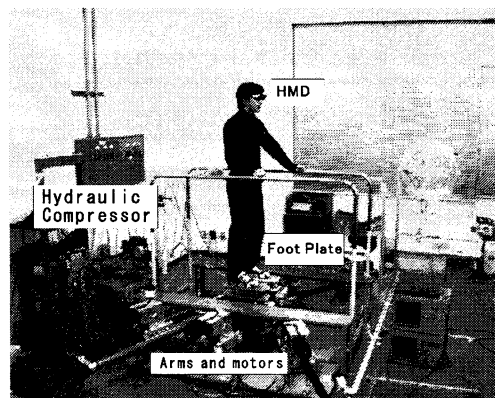
Figure 1: Virtual Walkway System

2   Virtual Walkway System

The Virtual Walkway System consists of a gait simulator and a virtual display system. Figure 2 shows the configuration of Virtual Walkway System. The virtual display system provides the user with various virtual space images through HMD.
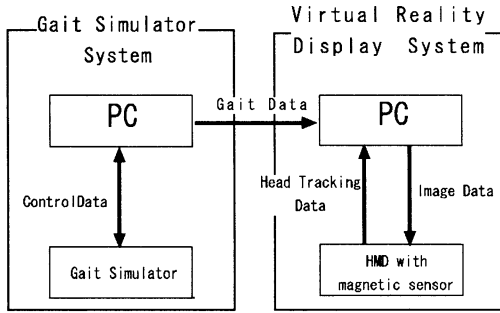
Figure 2: Composition of the Virtual Walkway System

3 Gait Simulator

The gait simulator has two foot plates and the user stands on foot plates. Figure 3(a) illustrates the action of foot plates when a user is walking straight on the simulator. The simulation procedure for straight walking is as follows;

- The right foot plate follows the right foot during swing phase of the right foot, and the left foot plate pulls back the left foot during stance phase of the left foot.

- All foot plates stop during stance phase of both feet.

- The left foot plate follows the left foot during swing phase of the left foot, and the right foot plate pulls back the right foot during stance phase of the right foot.

The user on the gait simulator doesn't lose his foot, because the foot plate is always under his foot during swing phase. The user freely stops, because all foot plates stop during stance phase of both feet.

Figure 3(b) illustrates the action of foot plates when going up stairs. The simulation procedure for going up stairs is as follows;

- The right foot plate follows the right foot in the horizontal plane and simultaneously rises up in the vertical plane during swing phase of the right foot. The left foot plate pulls back the left foot and descends in the vertical plane during stance phase of the left foot.

- All foot plates stop during stance phase of both feet.

- The left foot plate follows the left foot in the horizontal plane and simultaneously rises up in the vertical plane during swing phase of the left foot. The right foot plate pulls back the right foot and descends in the vertical plane during stance phase of the right foot.

Figure 3(c) illustrates the action of foot plates in changing of direction. When the user on this simulator changes of direction, the plate follows and pulls back the foot while rotating.

Figure 4 illustrates foot plates, rack-geared arms and PSD cameras. The gait simulator has two foot plates, these plates are controlled by three rack-geared arms and three AC motors change the length of these arms. Hydraulic cylinder mechanism is built under the foot plates and these plates can move up and down. Each motion range of the foot plate is 100cm in the forth and back direction, 20cm in the left and right direction and 20cm in the up and down direction, and is sufficient for normal human walking.

Two LEDs are set on user's shoes and their movements according to user's locomotion are monitored by two PSD cameras set in front of the gait simulator. As described before the LED position information is utilized for the control of rack-geared arms. Pressure sensors are also set on the shoe sole and these sensors monitor the phase of swing and stance of the user locomotion.
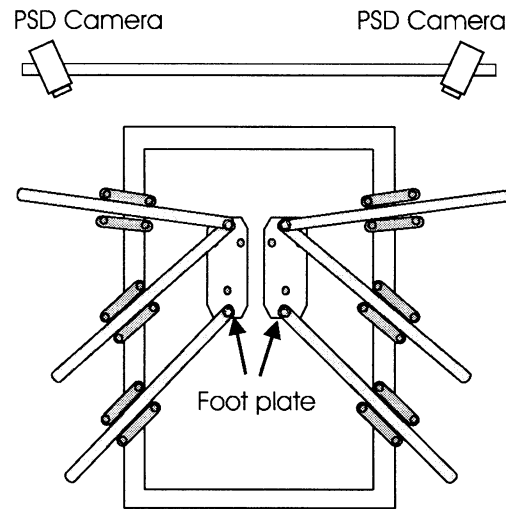
Figure 4: Foot plates, rack-geared arm and PSD cameras
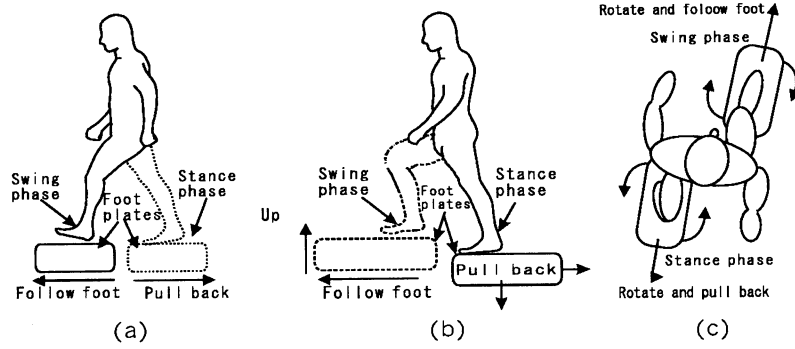
Figure 3: Operation principle

4 Virtual Reality Display System

In our Virtual Walkway System HMD (Glasstron, SONY) is used as the virtual reality display. A magnetometric sensor (Fastrak, POLHEMUS) is set on the HMD as a head tracking sensor.

The Virtual Reality Display System receives the locomotion data from the gait simulator and the head tracking data from the head tracking sensor. 3D virtual space moving images are generated by the Virtual Reality Display System using this information of user locomotion. Therefore the user can look at free direction of a virtual space.

We did not adopt the stereoscopic display. Therefore, the user should understand the position or the height of the stairs without the binocular parallax when he goes up the stairs. Of course this is a hard work for the user. Then user's foot is drawn in the virtual space.

Figure 5 shows an example of virtual space image. Figure 6 shows an example of virtual stairs and virtual feet.
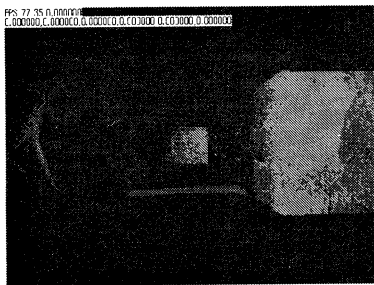
Figure 5: Virtual space image
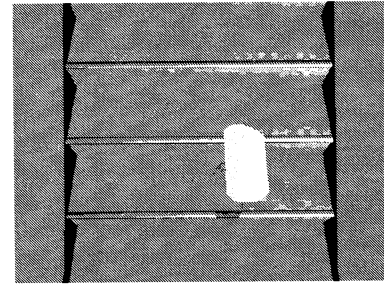
Figure 6: Virtual stairs and virtual foot

5 Evalution experiment and Discussion

Users with HMD are asked to walk on the gait simulator, enjoy virtual space walking and evaluate their walk on the gait simulator.

At first, we asked users to walk straight and the virtual space image was displayed in user's HMD. Next, we asked users to go up virtual stairs. Figure 7 shows sequential pictures of user locomotion during straight walking. As a result the user could walk at 3km/h, and could have an impression of walking in the virtual space.

Figure 8 shows sequential pictures during going up stair. The user could not go up stair without virtual feet because he didn't understand the position of each stair. However, when the virtual feet were displayed, the user could go up stair. Virtual feet help the user to understand his behavior in a virtual space.
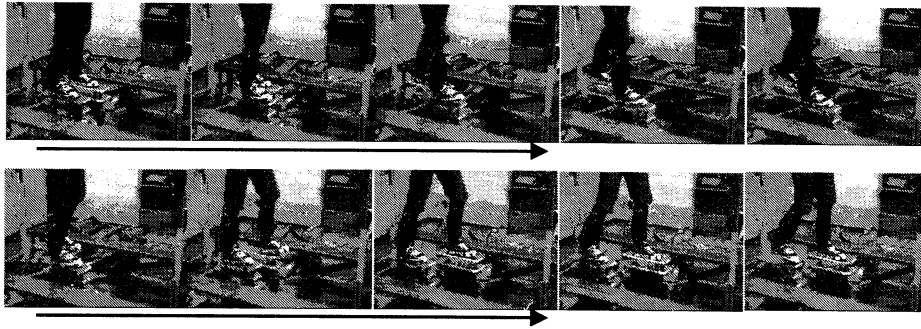
Figure 7: Sequential pictures (Straight walking)
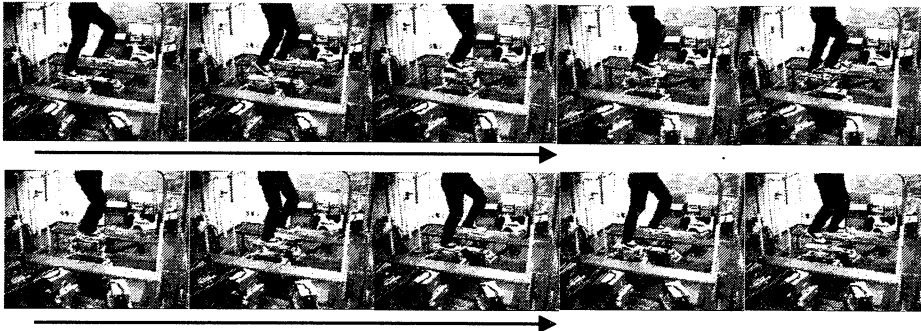
Figure 8: Sequential pictures (Going up stairs)

6   Conclusion

It was proved that the straight walking at 3km/h and going up stair on this system is possible in our Virtual Walkway System. For now we did not evaluate the direction change, because one foot plate clashed with other foot plate. We are now studying the way of direction change.

Any way this system has been proved to be effective for the simulation of various walking style. We are applying this Virtual Walkway System for rehabilitation and gait analysis.

References

[1] Darken, R. P., Cockayne, W. R. and Carmein, D., "The Omni-directional Treadmill ; A locomotion Device for Virtual Worlds", *Proc. UIST'97*, pp. 213-221, 1997.

[2] Iwata, H., "Walking About Virtual Environments on Infinite Floor", *Proc. IEEE 1999 Virtual Reality Annual International symposium*, pp. 286-293, 1999.

[3] Iwata, H. and Fujii, T., "Virtual Perambulator; A Nobel Interface Device for Locomotion In Virtual Environment", *Proc. of IEEE VRAIS'96*, pp. 60-65, 1996.

[4] Iwata, H., "Force displays for walkthrough simulation", *Proc. 2nd Intl. Symposium on Measurement and Control in Robotics*, pp. 481-486, 1992.

[5] Iwata, H., Yano, H. and Nakaizumi, F., "Gait Master: A Versatile Locomotion Interface for Uneven Virtual Terrain", *Proc. IEEE VRAIS'01*, pp. 131-137, 2001.

[6] Hirose, M. and Yokoyama, K., "Synthesis and transmission of realistic sensation using virtual reality technology", *Transactions of the Society of Instrument and Control Engineers*, 33(7), pp. 716-722, 1997.

Scaling Law in Common to Turbulence and Price Fluctuations

Mieko Tanaka-Yamawaki,
Department of Computer Science
and Systems Engineering
Miyazaki University
Miyazaki 889-2192 Japan
mieko@cs.miyazaki-u.ac.jp

Tsuyoshi Itabashi
Department of Computer Science
and Systems Engineering
Miyazaki University
Miyazaki 889-2192 Japan
cs9802@eagle.cs.miyazaki-u.ac.jp

Abstract

We study high-resolution data of foreign currency exchange rates and a newly taken data on the velocity fluctuations in three-dimensional turbulence in order to clarify the controversial issue of a proposed resemblance between turbulence and high-frequency exchange rates. It is found that both have scale invariance, both are fitted well by Lévy distributions of appropriate values of index α, and the moments follow the Kolmogorov scaling by assuming the analogy between {price, time, information} for market and {velocity, space, energy} for turbulence, though the indices of the two moment rules are shifted by a constant. For turbulence, the index ζ_n of n-th moment follows n/3+0.3, and for currency exchange market, the index ξ_n follows n/3+0.7.

1. Introduction

It is assumed in the conventional financial engineering that price fluctuations (of stocks or foreign exchange rates) follow the Gaussian statistics, which is equivalent to assume that each price movement is independent of the previous move. Although it is a convenient idealization for the sake of mathematical treatment, the real data show us various kinds of deviations from the perfect randomness.

The most notable point is the scale invariance, which introduces Lévy statistics instead of Gaussian. Mantegna and Stanley pointed out that the price increments of a stock index per minute (S & P500 for several years) actually follow the Lévy distribution of parameter α =1.4 [1].

Based on this, Ghashghaie, et. al. connected foreign currency markets to the world of turbulence [2]. It has been known that three-dimensional isotropic turbulence has a property of scale invariance, in the manner of Kolmogorov. It is characterized by the power-law behavior of the power spectrum S(f) as a function of frequency f. This region in f is called as the inertial range and the power (i.e. the slope in log-log plot) −5/3 is derived from a simple dimensional counting [3].

This further derives the scale-invariant behavior of moments (of velocity fluctuations) in the inertial range

$$\left\langle (\Delta v)^n \right\rangle = c\, (\Delta r)^{\zeta_n} \qquad (1)$$

and determine the index ζ_n to be n/3. (Kolmogorov's theory). It is known that Eq. (1) holds well for small n.

They compare the probability density distribution of very short-term foreign exchange rates and that of turbulent flow and confirm their similarity. Then they attempt to view a currency market in analogy to turbulence, by comparing the time in the market and the space in turbulence, information in the market and the energy in turbulence, and examine if a correspondent quantity as Eq. (1)

$$\left\langle (\Delta x)^n \right\rangle = c\, (\Delta t)^{\xi_n} \qquad (2)$$

holds or not in the market. They found that turbulence and currency exchange resemble very much in this context.

Stimulated by this work, we have done extensive numerical analysis on this line of research and, not only confirmed the resemblance between the two processes, but also found that the index ξ_n of foreign exchange rates stays longer on the slope 1/3 than the index ζ_n of turbulence [4].

In this paper, we discuss on this similarity based on our new analysis of foreign exchange data as well as newly taken data of turbulent flows.

We use two foreign exchange data FX1, FX2 and a turbulence data TRB. FX1 is the foreign exchange rates of U.S. dollar versus German mark from 1992 to 1993 having 1,470,000 data points. FX2 is the foreign exchange rates of U.S. dollar versus Japanese Yen from 1995 to 2000 having 9,800,000 data points. TRB is the data of turbulent flow having 250,000 data points taken at the turbulent wind tunnel located in Miyazaki University in May 2002.

2. Probability Density Distribution

We first calculate the probability density distribution function (p.d.f. hereafter) of price increments P(Δx) for various values of time resolution parameter Δt, and that of velocity increments P(Δv) for various values of space resolution parameter Δr. The price increment is defined as Δx =x(t)-x(t+Δt), and the velocity increments Δv is Δv =v(r)-v(r+Δr).

We can immediately recognize the similarity between P(Δx) in Figure 1 and P(Δv) in Figure 2. Both distributions are well fitted by Lévy distributions of slightly different indices, α =1.6 for Figure 1 and α =1.8 for Figure 2.
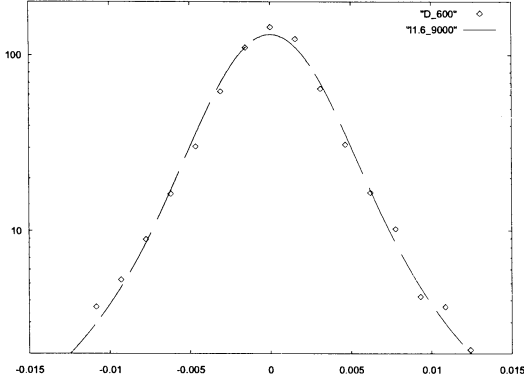
Fig. 1: Dots show the probability density distribution of price increments for time resolution $\Delta t = 600$, for foreign exchange data FX1. The vertical axis is $\log(P_{\Delta t}(\Delta x))$ and the horizontal axis is Δx. The dashed line shows the Lévy distribution of index $\alpha = 1.6$.
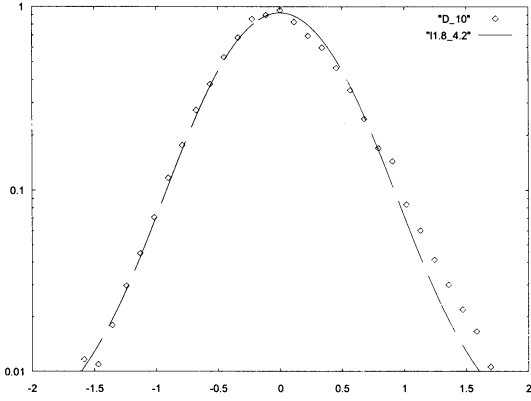
Fig. 2: Dots show the probability density distribution of velocity increments for space resolution $\Delta r = 10$, for turbulence data TRB. The vertical axis is $\log(P_{\Delta r}(\Delta v))$ and the horizontal axis is Δr. The dashed line shows the Lévy distribution of parameter $\alpha = 1.8$.

3. Lévy Distribution

3-1. Scale Transformation

As both $P(\Delta x)$ and $P(\Delta v)$ are fitted by Lévy distribution, we review some properties of this distribution. It is defined as

$$P_{\alpha,\beta}(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{\beta |k|^{\alpha} + ikx} dk \qquad (1)$$

It has scale invariance as follows.

$$P_{\alpha,\beta}(x) = \lambda^{1/\alpha} P_{\alpha,\lambda\beta}(\lambda^{1/\alpha} x) \qquad (2)$$

which means that $P_{\alpha,\beta}(x)$ at different values of β overlap each other by adjusting the scale. The scaling region is the range of λ within which the scale invariance holds.

Using Equation (2), we find the scaling region by changing the scale λ as long as they overlap on the same function. The scaling region thus obtained for FX1 is $\Delta t = 20$–5000 as in Fig. 3, for FX2 is $\Delta t = 30$–40000 as in Fig. 4, and for TRB is $\Delta r = 10$–70 for TRB as in Fig. 5.
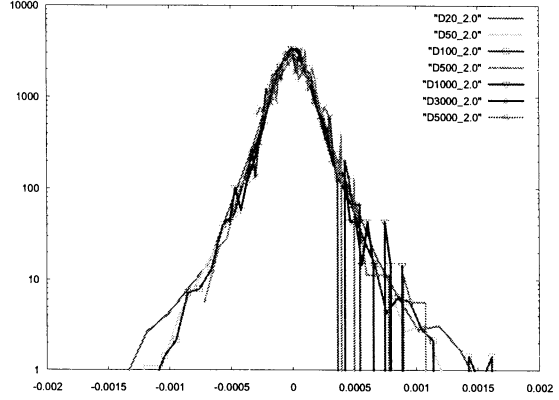
Fig. 3: The probability distributions of price increments $P_{\Delta t}(\Delta x)$ for various values of time resolution Δt (for data FX1) are shown to overlap each other by the scale transformation in Eq. (2).
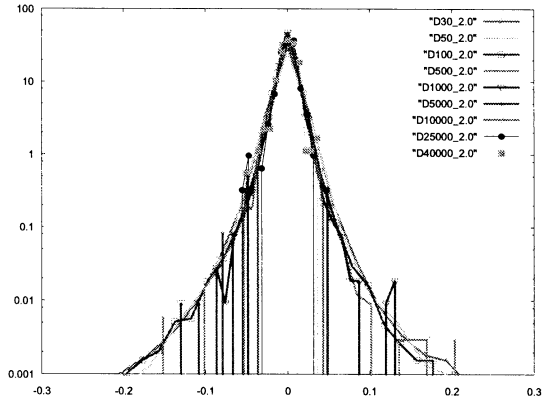
Fig. 4: The same as in Fig.3, for another data FX2.
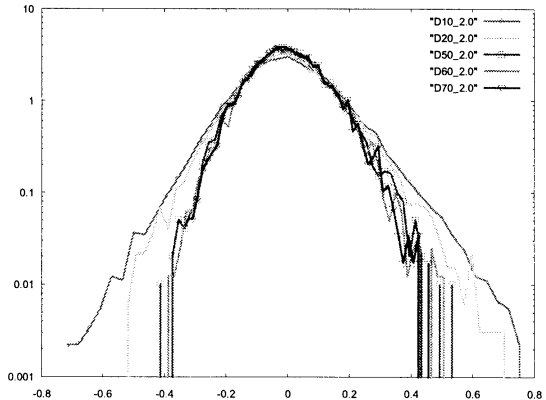
Fig. 5: The probability density distribution of velocity increments $P_{\Delta r}(\Delta v)$ for various values of space resolution Δr by using turbulence data TRB are shown to overlap each other by the scale transformation in Eq. (2).

3-2. Rate of Return to the Origin P(0)

If we set x=0 in Equation (2), we obtain

$$P_{\alpha,\lambda\beta}(0) = \lambda^{-1/\alpha} P_{\alpha,\beta}(0) \tag{3}$$

which provides us a simple way of identifying the scaling region by finding a region of a straight line in the log-log plot of P(0) vs. λ, and the index α is obtained from the slope of this line [1]. Figure 6, 7, 8 show the results of this analysis for FX1, FX2, and TRB, respectively.
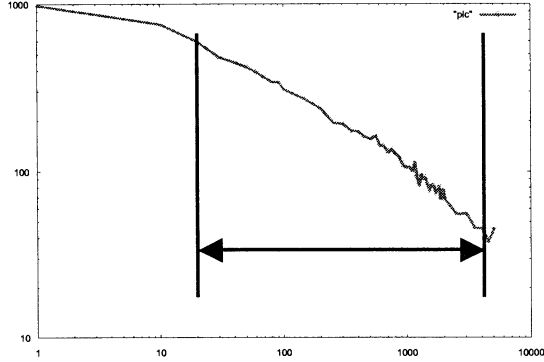
Fig. 6: The rate of return to the origin P(0) as a function of time resolution Δt is shown in log-log scale. The data used here is FX1. From the range of the straight line the scaling region is identified to be Δt=20 - 5000.
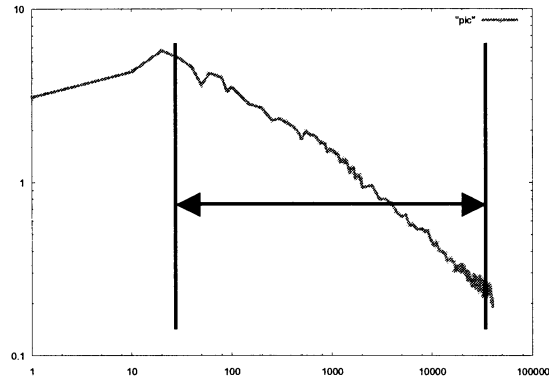
Fig. 7: The same plot as in Fig.6 for another data of currency exchange rate FX2. From the range of the straight line the scaling region is identified to be Δt=30 - 40000.
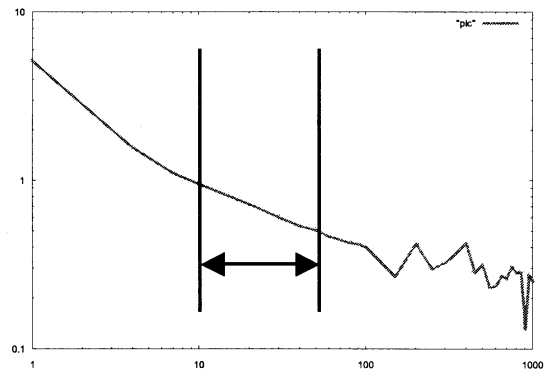
Fig. 8: The same plot as in Fig.6 for turbulence data TRB. From the range of the straight line the scaling region is identified to be Δr=10 to 70.

4. Moment Scaling Laws

Now we compute n-th moments as functions of the scale parameter λ and see whether the indices ζ_n and ξ_n in the scaling region can be fitted by n/3 as in Kolmogorov's theory.

Figures 9, 10, 11 show the n-th moments of FX1, FX2, and TRB, respectively. The index ξ_n of FX1 and FX2, and ζ_n of TRB are calculated by the least-square fit, and the results are shown in Fig.12, 13, and 14.

The index ξ_n of foreign exchange rates can be fitted by a line n/3+0.7 for n<3, and the index ζ_n of turbulent flow overlaps n/3+0.3 for n<2.
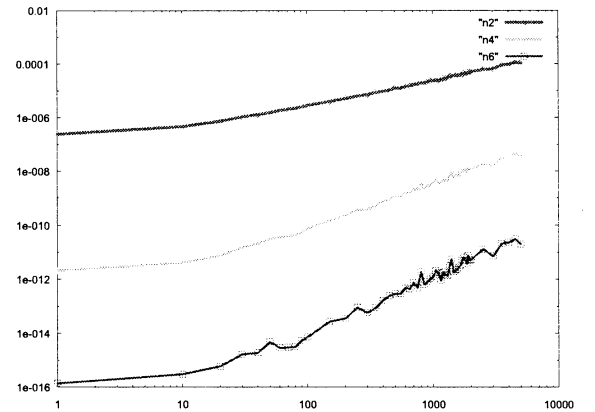
Fig. 10: The n-th moments of currency exchange rate FX1 for n=2, 4, 6. The vertical and the horizontal axis are $\log(\langle(\Delta x)^n\rangle)$ and $\log \Delta t$.
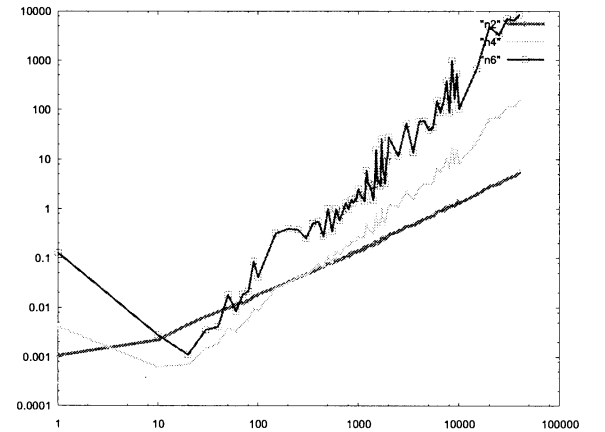
Fig. 11: The same quantities as in Fig.10 for FX2.
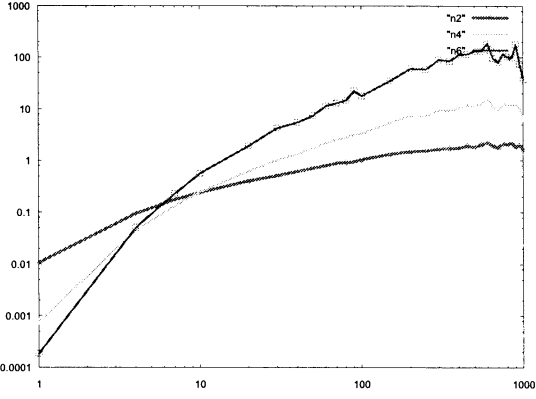
Fig. 9: The n-th moments of TRB for n=2, 4, 6. The vertical and the horizontal axis are $\log(\langle(\Delta v)^n\rangle)$ and $\log \Delta r$.
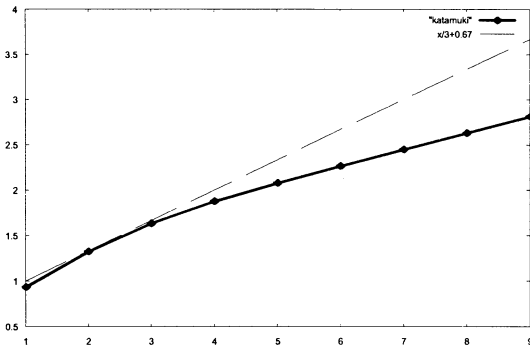
Fig. 12: The solid line shows the exponent ξ_n as a function of n for FX1. The dashed line is n/3+0.7.
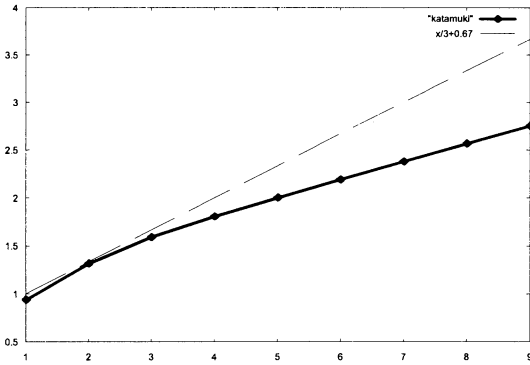
Fig. 13: The solid line is the exponent ξ_n as a function of n for FX2. The dashed line is n/3+0.7.
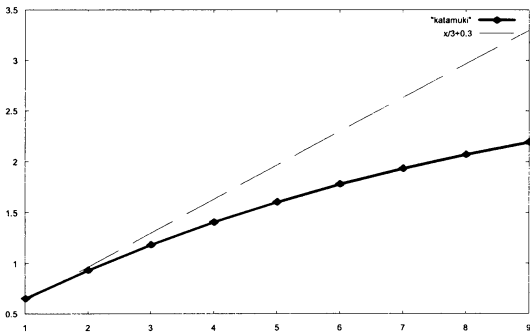
Fig14: The solid line shows the exponent ζ_n as a function of n for TRB. The dashed line is n/3+0.3.

5. Conclusion

We have examined the controversial issue of proposed similarity between turbulence and foreign exchange, by comparing the p.d.f. of appropriate quantities, and the indices of moment scaling laws. We have shown that both p.d.f. can be fitted by Lévy distributions for similar values of index α (1.6 for currency exchange and 1.8 for turbulence). Then we have computed the index ξ_n of foreign exchange rates and the index ξ_n of turbulent flow in the scaling region. It is found that the index ξ_n both for FX1 and FX2 overlap the line n/3+0.7 for n<3, and the index ζ_n for TRB overlaps the line n/3+0.3 for n<2. This result shows that both ζ_n and ξ_n has the same n-dependence, and both shifted by constant intercepts from n/3 derived in Kolmogorov's theory.

Acknowledgements

References

[1] Rosario N. Mantegna and H. Eugene Stanley, "Scaling behaviour in the dynamics of an economic index", NATURE, vol. 376, pp.46-49, 1995

[2] S. Gashghaie, W. Breymann, J. Peinke, P. Talkner & Y. Dodge, "Turbulent cascades in foreign exchange markets", NATURE, vol. 381, pp.767-770, 1996

[3] A.N. Kolmogorov, "The local structure of turbulence in incompressible viscos fluid for very large Reynolds number", Dokl. Acad. Nauk, SSSR, 30, 9-13, 1941

[4] Mieko Tanaka-Yamawaki, Tsuyoshi Itabashi, and Shinya Komaki, "Does Information Flow in High-frequency Financial Data as Energy Does in Turbulence", Proceedings of the 7th International Symposium on Artificial Life and Robotics, 1,pp.78-81, 2002

Characteristic Features of High Frequency Financial Time Series

Mieko Tanaka-Yamawaki
Department of Computer Science
and Systems Engineering
Miyazaki University
Miyazaki, 889-2192 Japan
mieko@cs.miyazaki-u.ac.jp

Shinya Komaki
Department of Computer Science
and Systems Engineering
Miyazaki University
Miyazaki, 889-2192 Japan
cs9815@eagle.cs.miyazaki-u.ac.jp

Abstract

Although financial time series are highly random, they have many characteristic features, especially in the high frequency data (tick data). We expect, therefore, to utilize those deviations from the perfect randomness for the possibility of prediction. We have found so far an extreme regularity in the conditional probabilities observed over years, and the meaningful memory length is about two or three ticks, based on an extensive analysis of large volume data. We also discuss on the minimum size of the data for computing conditional probabilities.

1 Introduction

It is widely believed that the financial time series are maximally random and fine movements of prices behave as the perfect random walk. This leads to a conventional view of financial markets being efficient, thus no arbitrage chance exists. However, there have been many evidences suggesting non-Gaussian statistics [1], imperfect random walks, and existence of arbitrage chances [2] (Arbitrage means to take a profit by using unequal prices at the same moment.) for price dynamics based on the real data analysis of short-term fluctuations.

We first examined how much arbitrage chance is observed in the triangle trades among three currencies, U.S. Dollar, German Mark, and Japanese Yen, by using the test data called HFDF93 (High Frequency Data in Finance), taken for one year from 1992 to 1993. We have found that almost ten percent of the total data corresponds to arbitrage chance, which means we could in principle make a profit just by exchanging from USD to DEM, from DEM to JPY, and JPY to USD, or to the opposite direction to this order. Although most of such chances quickly disappear, we found that some continues for a very long time, and the average relaxation time of such a chance is 6 seconds, once it occurs. [3]

Knowing that the very fine motions of the financial time series are not perfectly random and arbitrage chances indeed exist, we seek for more explicit knowledge on the behavior of financial time series in order to predict when such chance may occur.

It was first pointed out by Ohira, et. al. [4] that certain conditional probabilities are almost identical in two distinct sets of tick data of foreign exchange separated over some months, if we neglect unmoved ticks and focus only on the directions of up and down motions. Those conditional probabilities are thus represented by binary numbers, $\{0,1\}=\{down,up\}$, and there are 2^m conditional probabilities when we consider m steps of memory.

We have examined this property in more systematic way [5]. We have analyzed a large continuous data of USD/JPY exchange rate ('ask' position) extending for 5+1/2 years from 1995 to 2001 having more than ten million ticks, by automatically generating all the conditional probabilities in the computer and found that those binary conditional probabilities are indeed very stable for the entire range of this data.

For example $P(1|0)$ stays around 0.71, and $P(1|00)$ also stays around 0.78, with very small variances. This stability keeps for memory depth up to 4, and becomes worse for m > 4. We have also attempted to use this knowledge for predicting the future time series (in one tick).

In this paper, we report some of our new findings in this line of research. We use the same data as before, and examine how far the same feature holds if we change the size of a piece of data over which we compute the conditional probabilities, and also investigate if the conclusion is consistent with other results such as the length of auto-correlation, and the behavior of mutual

information.

2 Conditional Probabilities

In this section, we show our result of the conditional probabilities for the case of binary fluctuations {up, down}={1,0} neglecting unmoved data points. We have examined the regularity of the conditional probabilities in the following way. First we split the 10,000,000 data points into pieces to calculate the conditional probabilities. We then compute all the 2^m conditional probabilities for each data in the case of memory depth m. At this point, we do not know what is the appropriate size we should split. We have tried various sizes and compare the variances. Figure 1 shows the result of this analysis on the conditional probability $P(1|0)$.
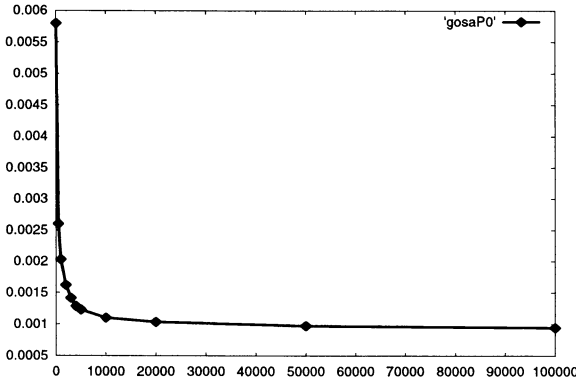
Fig. 1: Stability of the conditional probability $P(1|0)$ is represented by the mean squared errors as a function of the data size over which $P(1|0)$ is calculated. MSE reaches the stable point when the size exceeds 20,000.

We can see that the result is stable if the size of each piece to compute $P(1|0)$ exceeds 20,000. Choosing the size 50,000, we split the total 10,000,000 data points into two hundred sets, each of which contains 50,000 data points. We then compute all the 2^m conditional probability for 200 data sets for memory depth m=1-4. Figures 2 - 5 show the conditional probabilities for memory depth m=1- 4.

These graphs tell us the following things. If price fluctuations were perfectly random, i.e., each step of a time series is independent of its history. In this case we expect $P(1|0)=P(1|1)=1/2$. Figure 2 clearly shows that this is not the case. Our result shows that the motion to the opposite direction is much larger than the

motion to the same direction, $P(1|0)=0.71$ >> $P(1|1)=0.29$. Figure 3 tells us a benefit of taking account of two steps of the history. Namely, $P(1|0)$ splits into $P(1|00) = 0.78 \pm 0.05$ and $P(1|10) =0.68 \pm 0.03$, which indicates we can get more information by considering longer history. Figure 4, however, tells us that it is of no use considering longer history than m=2. For example, $P(1|00)$ does not split into $P(1|000)$ and $P(1|100)$ any more. In fact eight conditional probabilities are bunched into four groups. This situation is more clearly shown in Figure 5 for the case of m=4, where sixteen conditional probabilities are again bunched into four groups. Thus we conclude that there is no need to consider m > 2 according to Figures 2-5.

As a result we have confirmed that memory depth is two or three as described in Ref. [5].
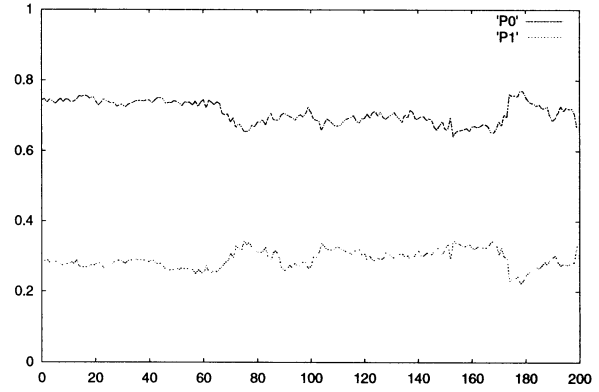
Fig. 2: Conditional probabilities of depth one, $P(1|A)$ where A is 0 or 1 is plotted as a time series.
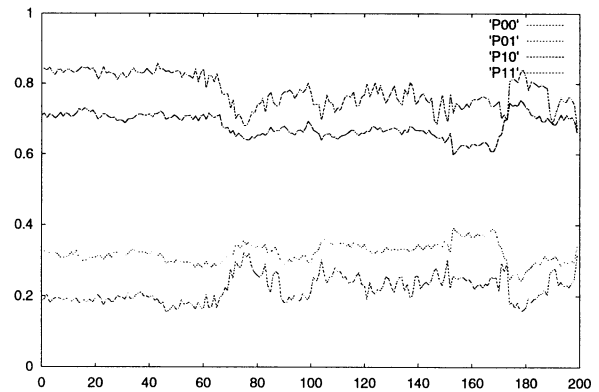
Fig. 3: Conditional probabilities of depth two, $P(1|AB)$, where A and B are binary numbers, are plotted as times series.
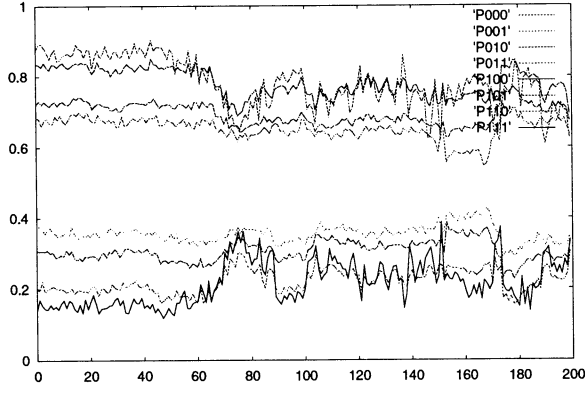
Fig. 4: Conditional probabilities of depth three, P(1|ABC), where A,B, and C are binary numbers, are plotted as time series.
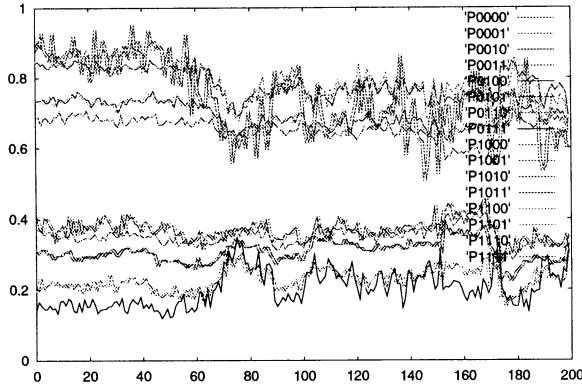
Fig. 5: Conditional probabilities of depth four, P(1|ABCD) where A,B,C, and D are binary numbers, are plotted as time series.

3 Memory Depth of Auto Correlation and Mutual Information

We have so far confirmed that the memory depth m < 3 is relevant to predict the future. In this section we consider the relevant memory depth by two other methods, the auto-correlation and the mutual information.

Auto correlation of the price x is define as follows:

$$C(T) = \frac{<\Delta x(T+t)\Delta x(t)> - <\Delta x(T+t)><\Delta x(t)>}{\sigma^2}$$ (1)

where Δx is the price differential of foreign exchange rate and σ is its standard deviation. C(T) vanishes if Δx (t) and Δx (t+T) are uncorrelated. C(T) computed for the currency exchange data is shown in Figure 6, which vanishes quickly after T=3 or 4. We conclude,

therefore, that the relevant memory length in C(T) is also 3 steps or around.
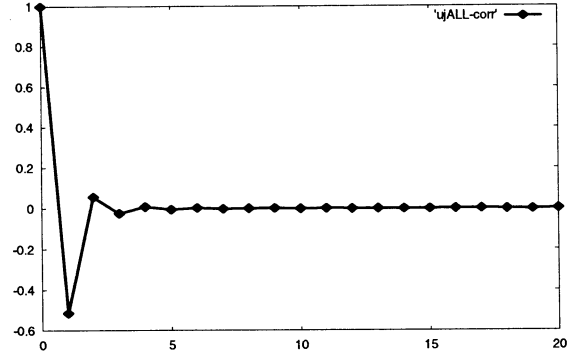
Fig. 6: Auto-correlation C(T) as a function of T for foreign exchange data explained in Section 1 vanishes at around T=3 or 4.

Next we analyze the same issue by using the mutual information. The mutual information M(x,y) is defined as follows:

$$M(x,y) = H(x) - H(x|y)$$
$$= -\sum_x P(x)\log_2 P(x) + \sum_x \sum_y P(x|y)\log_2 P(x|y)$$ (2)

In using Eq.(2), we have computed P(x) and P(x|y) by using the initial 52,000 ticks out of the 10 million data points. Namely, 1-50,000 points are used to obtain the initial points of P(x) and P(x|y), and 2,001-52,000 are used to obtain the final point of P(x) and P(x|y). M(x,y) thus computed have 2,001 points. In Figure 7, four lines of M(x,y) are plotted. The line at the bottom corresponds to m=1 (y=0 or 1), and the line next to the bottom corresponds to m=2 (y=00,01,10,or 11), the two lines almost overlapping are the ones for m=3 (y=000,..., 111) and m=4 (y=0000,..., 1111).

We can see that M(x,y) for m=1 and that for m=2 are well separated each other, and M(x,y) for m=3 is also distinct from the previous two lines. However, the line m=4 is indistinguishable from the previous line. This fact tells us that the history longer than m=3 does not give us extra information for the sake of predicting the direction of the future move.
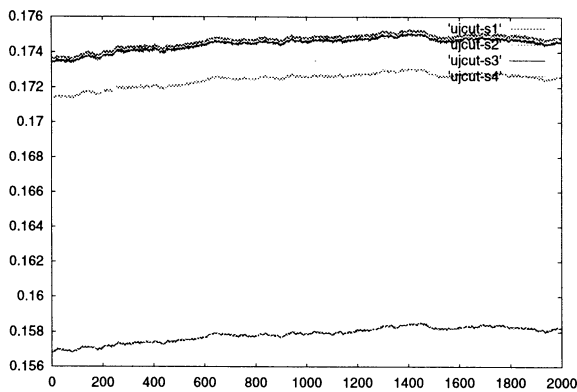
Fig. 7: Time series of mutual information M(x,y) obtained in the way explained in the text.

4 Conclusion

We studied in this paper tick data of foreign exchange rate of US dollar versus Japanese Yen from 1995 to 2001 having 10,000,000 data points. We have confirmed the previous result on the conditional probabilities being very stable over several years. We also determined the effective length of memory by means of three different ways. It is found that the auto-correlation disappears after 2 or 3 ticks, and mutual information between the next motion and the past m motions for m > 3 overlap the same quantity for m=3, indicating that the memory for more than three does not give us any extra information.

Acknowledgements

References

[1] R.N. Mantegna and H.E. Stanley, "Scaling Behavior in the Dynamics of an Economic Index" Nature, 376, 46-49, 1995.

[2] Y. Aiba, et. al., "Triangular Arbitrage as an Interaction Among Foreign Exchange Rates", Physica A 310, 467-479, 2002

[3] Mieko Tanaka-Yamawaki and Shinya Komaki, "A Numerical Analysis in the Triangular Arbitrage Chances", Proceedings of 2002 International Conference on Nonlinear Theory and Its Applications, pp.415-418, 2002

[4] Toru Ohira, et.al. "Predictability of Currency Exchange Market ", Physica A 308, 368-374, 2002

[5] Mieko Tanaka-Yamawaki "A Study on the Predictability of High-frequency Financial Data", Proceedings of the 7th International Symposium on Artificial Life and Robotics, vol.1, pp.74-77, 2002

Autonomous Learning of Reward Distribution in Not100 Game

Katsunari Shibata(shibata@cc.oita-u.ac.jp), Tsutomu Masaki & Masanori Sugisaka

Dept. of Electrical & Electronics Engineering, Oita University, 700 Dannoharu, Oita 870-1192, Japan.

Abstract

In this paper, autonomous learning of reward distri-
bution in multi-agent reinforcement learning was ap-
plied to the 4 player game named "not100". In this
game, more shrewd tactics to cooperate with the other
agents is required for each agent than the other tasks
that the learning was applied previously. The reward
distribution ratio after learning was varied among sim-
ulation runs. However, the validity of the average
non-uniform reward distribution ratio was examined
in some ways. The three agents with higher win proba-
bility after learning cooperated mutually, while strong
cooperation was not observed in some cases when the
agents learned with a fixed distribution ratio.

1 Introduction

In multi-agent systems, since it is difficult to know
the policy to solve a given task in advance, au-
tonomous learning such as reinforcement learning is
useful. However, it is one of the difficult problems to
decide "reward distribution among agents" that af-
fects to generate cooperative behaviors. There are
some methods proposed already, but the reward dis-
tribution ratio is decided in advance. For example,
one agent gets the whole reward, or the reward is dis-
tributed uniformly to all the agents. There is also the
method that the agent who contributes directly to get
a reward gets a part of the reward, and the rest of it
is distributed to the other agents uniformly[1]. How-
ever, since appropriate distribution depends deeply
on a given task, enough knowledge about the task is
required to determine the distribution. Accordingly
there is much possibility that this spoils the effective-
ness of reinforcement learning, which is useful when
the knowledge about the given task is not enough.

Then the method has been proposed that the agent
learns the reward distribution ratio to the other agents
together with the learning of actions[2]. The learning
is based on the principle that by the distribution of
the reward, the other agents become to help him, and
finally he gets more reward or gets the reward earlier.
This method was applied to a simple two-agent and
three-agent competitive problems. It was shown that
(1) the state that no agents get any rewards by a con-
tinuous conflict can be avoided, (2) the distribution
ratio becomes small to the agent whose help is not
needed, and (3) not only the ratio but also the change
of the ratio affects the action learning. In the three-

agent problem, the tactics of an agent was observed
such that he disturbs the goal of another agent who
gave him less reward than the other one.

In this paper, the reward distribution learning is ap-
plied to a different type of problem, and the effective-
ness in wide area is verified. The problem has different
properties as (1) more shrewd tactics is required, (2)
some agent always can get reward in a finite time, (3)
the time to get the reward is not considered, and (4)
not only one agent can get reward. The problem is
the four-player game named "Not 100 game". In this
game, since one player always loses the game when the
other three players cooperate mutually, shrewd tactics
is necessary. It was verified whether cooperation with
the other agents emerges, and whether the distribution
ratio obtained by the learning is valid or not.

2 Learning of Reward Distribution

The reward distribution is learned together with the
action, but the time scale is different between them.
The action is learned at each time step, while the re-
ward distribution ratio is fixed for some trials, and is
updated according to the total reward obtained dur-
ing the period when the ratio is fixed. Concretely,
$dist_{ji}$ that is the reward distribution ratio from agent
i to agent j is changed using random numbers, and is
fixed for one cycle that is defined as N trials. Since
the distribution ratio should always satisfy

$$\sum_{j=1}^{A} dist_{ji} = 1.0, \qquad (1)$$

the change of the ratio $\Delta dist_{ji}$ is calculated as

$$\Delta dist_{ji} = rnd_{ji} - rnd_{(j+1)\%A,i}, \qquad (2)$$

where A: the number of agents and rnd: a random
number. When $dist$ becomes larger than 1.0 or less
than 0.0, it is set to be 1.0 or 0.0 respectively. The
difference is distributed uniformly to the ratio to the
others. In order to remove the effect in the transition
period, the total reward R in the latter half of the
cycle is calculated using the reward r_i of agent i as

$$R_i = \sum_{n=N/2+1}^{N} \sum_{i=1}^{A} dist_{ji} r_i(n) \qquad (3)$$

and is evaluated. The learning is so simple that when
R is larger than the previous value, the distribution
ratio is set as the default value, and otherwise, the
distribution ratio is restored to the previous value.

Here, each agent acts sequentially, and the state transition is deterministic. The action learning is based on Q-learning. Since the time to get the reward is not necessary to be considered in the problem in this paper, the discount factor γ is set to be 1.0. As an example, the learning of the agent $j = 0$ is shown in the followings. The state evaluation $V_j(s_j(t+1))$ just after his action is calculated from the possibility $Pfin$ that the game finishes before his next tern, expected reward in that case $\bar{r}fin$, and the expected maximum Q-value $maxQ_j$ at his next tern as

$$V_j(s_j(t+1)) = Pfin_j(s_j(t+1))\bar{r}fin_j(s_j(t+1))$$
$$+(1 - Pfin_j(s_j(t+1)))maxQ_j(s_j(t+1)). \quad (4)$$

Each term on the right hand side is calculated as

$$Pfin_j(s_j(t+1)) \leftarrow (1 - \alpha)Pfin_j(s_j(t+1)) + \alpha$$
$$\text{if the game finishes before his next tern}$$
$$\leftarrow (1 - \alpha)Pfin_j(s_j(t+1))$$
$$\text{otherwise}, \quad (5)$$

$$\bar{r}fin_j(s_j(t+1)) \leftarrow (1 - \alpha)\bar{r}fin_j(s_j(t+1))$$
$$+\alpha \sum_{i=0}^{A} dist_{ji}r_i(t+k)$$
$$\text{if the game finishes at } t+k, \quad (6)$$

$$maxQ_j(s_j(t+1)) \leftarrow (1 - \alpha)maxQ_j(s_j(t+1))$$
$$+\alpha max_k(Q_j(s_j(t+A), a_k)) \quad (7)$$

where α: a learning constant. Q value is learned using V as

$$Q_j(s_j(t), a(t)) \leftarrow (1 - \alpha)Q_j(s_j(t), a(t))$$
$$+\alpha \sum_{i=0}^{A} dist_{ji}r_i(t+1) + V_j(s_j(t+1)) \quad (8)$$

3 Not100 Game

Here, not 100 game is introduced. As shown in Fig. 1, 4 players sit at a table, each player counts within 3 numbers sequentially, and the player who counts 100 loses the game. Fig. 2 shows the rule of this game.

In this game, one player cannot win the game when the other 3 players cooperate mutually. Accordingly, it becomes important how to get the help of the other players to win the game. Actually, the game is interesting at the point that the human relation between players and the character of each player can be peeped.

Fig. 3 shows two examples of the game processes. Here, it is supposed that the player A, B, and C cooperate mutually. If the player D counts 97, 98 or 99, one of the others has to count 100, and the player D can win the game. If he counts one of the numbers from 90 to 96, since one player can count one, two or three numbers at one time, he has to count 100 at the next tern as shown in the upper process in Fig. 3. However, when he counts 89, he also cannot count 97, 98,

Figure 1: Not 100 game

Figure 2: The rule of "Not 100 Game"
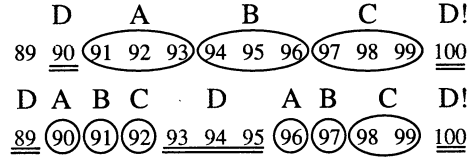
Figure 3: The reason why the agent always loses the game when the other 3 agents cooperate mutually

or 99 at the next tern as shown in the lower process in Fig. 3, he has to count 100. At the case of Fig. 1, the loser depends on whether the player C counts only 98 or two numbers of 98 and 99.

In this paper, to make the computation time short, "Not 30 game" is employed on behalf of "Not 100 game". To all the agents except for the agent who counts 30, the reward 1.0 is given. In the previous tasks to which the authors applied the learning, it is clear that the reward distribution is profitable for any of the agents, because no reward is given when the conflict state happens. However, in this problem, three agents always get the reward in a finite time, and the appropriate distribution is not clear.

4 Simulation

One cycle is defined as $400(= N)$ trials, and the distribution ratio is updated at every cycle. 10000 cycles are done in one simulation. The initial distribution ratio is decided randomly with the condition that each

——— self reward distribution ratio ——— win probability

(A) when reward distribution ratio was learned

(B) when reward distribution ratio was fixed at 0.91 for itself and 0.03 for the others
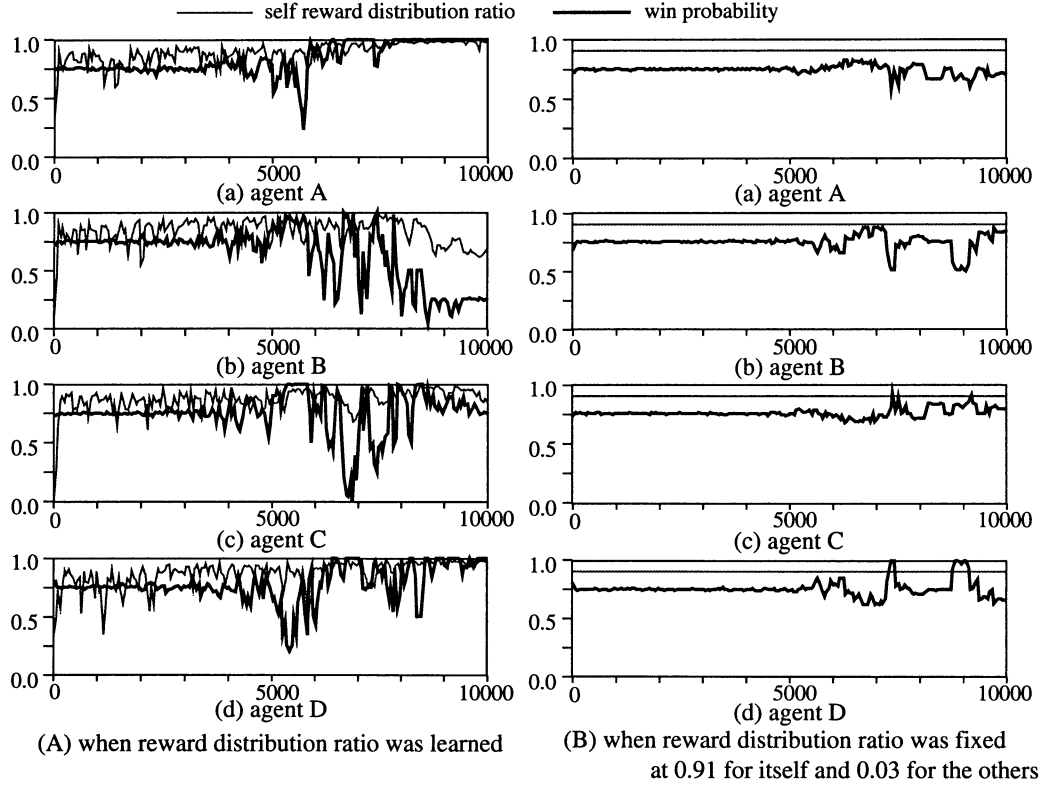
Figure 4: Change of each agent's win-probability and reward self-distribution ratio

one is positive and the total is 1.0. The range of the random number added to the distribution ratio is reduced from 0.1 to 0.01 linearly in log scale in 9000 cycles. The first count agent is decided randomly at each trial. The Boltzmann selection is employed for the action selection after normalizing to make the maximum Q value become 1.0. The temperature is also reduced from 1.0 to 0.1 linearly in log scale in 8000 cycles.

Fig. 4(a) shows the change of the self reward distribution ratio and the win probability for each of 4 agents. In the early stage of the learning, since the temperature was high, the win probability of every agent was almost 0.75. The self-distribution ratio became around the value from 0.8 to 0.9 soon even if the initial ratio is decided randomly. The win probability began to fluctuate from around 5000 cycle, but the way of change varied depending on the simulation run. The distribution ratio of the agent whose win probability is large is apt to be large.

For comparison, Fig. 4(b) shows the result when the distribution ratio is fixed at 0.91 for himself and 0.03 for the others. The fluctuation of the win probability is smaller. The reason can be thought that in the learning case, the small change of the distribution ratio sometimes influences the win probability.

Next, the validity of the distribution ratio is verified. Table 1 shows the average ratio after learning over 100 simulation runs. It is seen that the self-distribution ratio of the agent whose win probability is

Table 1: The reward distribution ratio after learning.

the agent whom the reward is distributed to			
	myself	next agent	opposite agent	previous agent
the agents with win_prob>0.9	0.961	0.011	0.013	0.014
all agents	0.895	0.037	0.033	0.035

large becomes large. Even though the ratio varied actually, it is also seen that depending on the simulation run, the ratio to the next agent is slightly smaller than the other agent. The reason can be thought that the help of the next agent contributes the win probability less than the other agents.

Furthermore, the ratio of one agent was fixed, and the win probability, total reward, and the average self-distribution ratio of the other agents were observed. Fig. 5 shows them as a function of the fixed ratio. Each of them is the average over 10 simulation runs. When the fixed ratio is small, the win probability is 1.0 except for the case of 0.0. It is suddenly decreasing as the fixed ratio becomes larger around 0.9. The total reward is the maximum when the fixed ratio is around 0.8. The acquired self-distribution ratio after learning as in Fig. 1 is larger than 0.8, but is not different so much. It is interesting that the self-distribution ratio of the other agents also becomes larger when the fixed ratio is larger than 0.8. This is because even
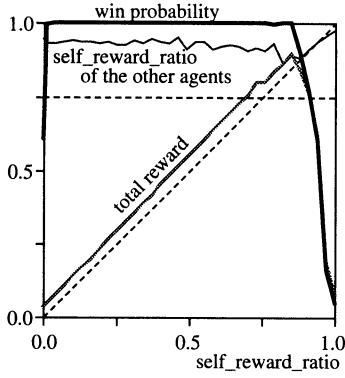
Figure 5: The win probability, total reward, and self-distribution ratio of the other agents as a function of fixed self-distribution ratio.
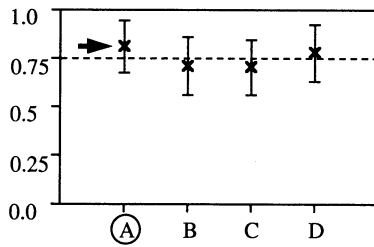
(a) the reward is distributed uniformly to the other agents in agent B, C, D
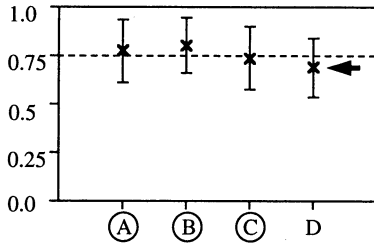
(b) the reward is distributed uniformly to the other agents in agent D

Figure 6: Effect of non-uniform reward distribution. The alphabet with circle indicates the agent with non-uniform reward distribution.

though the other agents makes their self-distribution ratio large, they can win the game.

The validity of the non-uniform distribution to the other agents as shown in Table 1 is verified. Fig. 6 (a) shows the win probability when the ratio of the agent A is fixed as Table 1 and that of the others is fixed such that the self-distribution is the same but the rest is distributed uniformly to the others. The win probability of the agent A is larger than the others. As shown in Fig. 6(b), when only the agent D distributed uniformly to the others, the win probability of the agent D became smaller. From these results, the obtained weighted ratio is supposed to be valid.

Finally, the learning was performed in 4 groups, and the win probability was observed when the weakest

Table 2: Change of win probability by the replace of one agent between two groups.

			Group A (receiving)			
		max	learning	selfish 1.0-0.0	0.91 -0.03	uniform 0.25-0.25
	min		0.238	0.578	0.592	0.711
Group B (sending the strongest agent)	learning	1.000		0.504	0.511	0.898
	selfish	0.921	0.158		0.568	0.932
	0.91 -0.03	0.889	0.175	0.554		0.920
	uniform	0.784	0.096	0.302	0.345	

agent was replaced by the strongest agent in the other group. In one group, all the agents learned the ratio. In other groups, the ratio of every agent was fixed. The fixed ratio varied among three groups as in Table 2. The result are shown in Table 2. When the weakest agent in the learning group was replaced, the win probability of the newcomer agent is only around 0.1, while the win probability of the strongest agent in the learning group is more than 0.5 in every other group. However, the strongest agent in the selfish group or the group of (0.91-0.03) ratio could win more often than the strongest agent in the learning group. The result can be interpreted that in the learning group, an appropriate cooperation strategy is obtained by the agents, it is hard for the other group agent to win. However, since the cooperation strategy is not effective in the other groups, the strongest agent in the learning group could not win very much.

5 Conclusion

The reward distribution learning was applied to "Not 100 game". The weighted reward distribution was observed, and the validity was examined in some ways. The cooperation could be observed when the distribution ratio was learned, while it could not be observed when the distribution ratio is fixed as to distribute the reward to the other agents uniformly.

References

[1] Shirakawa, H., et al., "Experimental Study on Emergence of Cooperative Action Using Reinforcement Learning", Proc. of the 5-th Intelligent Systems Symp., pp. 119–124, 1998. in Japanese

[2] Shibata, K. and Ito, K., "Autonomous Learning of Reward Distribution for Each Agent in Multi-Agent Reinforcement Learning", Intelligent Autonomous Systems, 6, pp. 495–502, 2000.

Evolutionary and Time-Varying Reinforcement Learning System for Unobservable Dynamic Environment

Kosuke Umesako
Graduate School of Sci. and Eng.
Yamaguchi University
Ube, Yamaguchi 755-8611

Masanao Obayashi
Faculty of Eng.
Yamaguchi University
Ube, Yamaguchi 755-8611

Kunikazu Kobayashi
Faculty of Eng.
Yamaguchi University
Ube, Yamaguchi 755-8611

Abstract

We propose an evolutionary reinforcement learning (RL) system with time-varying parameters. The proposed system has three characteristics: 1) It can deal with an unobservable dynamic environment by using time-varying parameters; 2) The division of state space is acquired evolutionarily by genetic algorithm (GA); 3) One does not have to design the rules constructing an agent in advance. The efficacy of the proposed system is demonstrated by the control of a mobile robot in unobservable dynamic environment through computer experiments.

1 Introduction

So far many RL systems have been proposed [1, 2, 3]. Those systems adjust constant or non time-varying parameters; using those systems it is difficult to acquire appropriate behavior in complex and dynamic environment. The learning becomes more difficult if the variation of the environment cannot be observed. Hence, we propose the RL system whose parameters can vary with time. The proposed system can deal with the variation of the environment by varying parameters even if its variation cannot be observed. In addition, we propose a new division method of state space using GA. It is important problem to divide the state space of the environment appropriately. In the dynamic environment, it is difficult due to especially the variation of the environment. However, the division of the state space is acquired evolutionarily in the proposed system. Hence, using the proposed method a designer does not have to decide the division of the state space.

The efficacy of the proposed system is demonstrated by the control of a mobile robot in unobservable dynamic environment through computer experiments. The robot runs on a floor surrounded with walls toward a goal; there are some moving walls on the floor.

Since the position of walls is not given to the robot, the robot cannot observe the variation of the environment. It is shown that the proposed system can acquire the behavior that arrives at the goal in the unobservable dynamic environment.

2 Reinforcement learning

RL [1] is the learning method based on punishment or reward, and the aim of the learning is maximization of total rewards obtained during trials. An agent observes a state of environment and determines an action stochastically; it learns based on rewards obtained from environment. As a result, the RL system can search better action; it is called trial and error. In the proposed method, we use the stochastic gradient ascent method (SGA) [2] for RL method, since it can deal with POMDPs (Partially Observable Markov Decision Processes).

3 Evolutionary and time-varying reinforcement learning system

The proposed RL system is shown in Fig. 1. The system is constructed by rule part and action-selection part. The rule part estimates the appropriate parameters used for action selection, and it learns based on rewards obtained from environment. This part has time-varying parameters and the division of state space is carried out in this part. The action-selection part selects an action stochastically based on these estimated parameters. One can use arbitrary action-selection method for this part.

3.1 Rule part

This part is shown in Fig. 2. There are many rules in this part. Each rule has parameter vector used for
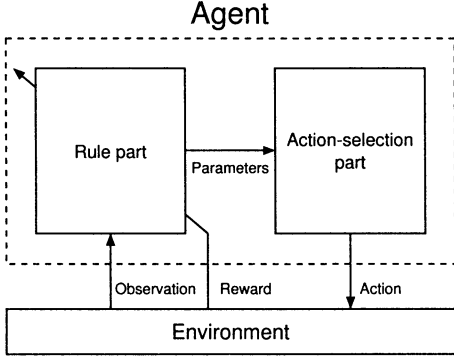
Figure 1: The proposed system

Figure 2: The rule part

action selection; these vectors are weighted by time-varying parameters. Each rule is in charge of a part of the state space of environment; it is determined by gene; it evolves by GA. A rule is described by:

$$R^k : \quad \text{if } (x_1 \text{ is } B_1^k, \ x_2 \text{ is } B_2^k, \ \cdots, \ x_n \text{ is } B_n^k)$$
$$\text{then } \boldsymbol{p}^k = (p_1^k, p_2^k, \cdots, p_m^k) \tag{1}$$

where $x_1, x_2, \cdots, x_n$ are the elements of the environment's state vector $\boldsymbol{x}$, $B_1^k, B_2^k, \cdots, B_n^k$ are crisp sets determined by gene, $\boldsymbol{p}^k$ are the parameter vector used for action selection (they are learning parameters), and m is the number of action's variety. The estimated parameter vector for action-selection part $\boldsymbol{p} = (p_1, p_2, \cdots, p_m)$ is calculated by:

$$p_j = \frac{\sum_{k \in A} w^k(t) p_j^k}{\sum_{k \in A} w^k(t)} \tag{2}$$

where A is the set of rules whose condition part are satisfied, $w^k(t)$ is a time-varying parameter calculated by:

$$w^k(t) = \left\{ \sin(w_1^k t + w_2^k) + 1 \right\} / 2 \tag{3}$$

where w_1^k, w_2^k are learning parameters. Since the time-varying weights are used, the influence of a rule varies. Hence, it becomes possible to share time between rules; the proposed system can easily deal with a dynamic environment.

3.2 Action-selection part

We use the roulette selection using Boltzmann distribution. For example, the probability of the selection of the jth action a_j is calculated by:

$$P(a_j | \boldsymbol{x}) = \frac{\exp(p_j / T)}{\sum_{j=1}^{m} \exp(p_j / T)}, \tag{4}$$

where T is temperature.
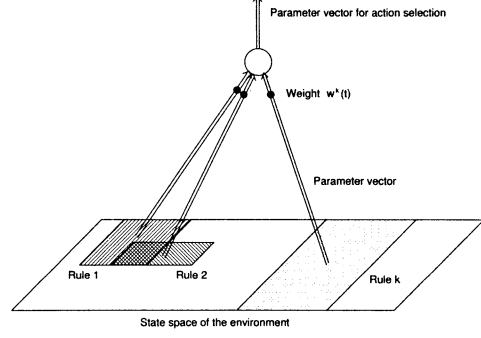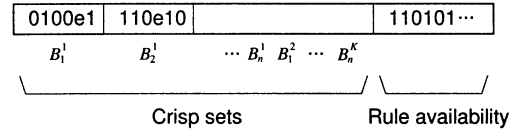
Figure 3: An example of the gene

3.3 Evolution of rules

In the proposed system, the condition part of rules is determined by gene and it evolves by GA. An example of the gene is shown in Fig. 3. For each crisp set, a bit string that has the identical length with each other (the length is 6[bits] in Fig. 3) is allocated. '0' corresponds to the lower half of the environment's state space; '1' corresponds to the upper half. A region in charge of a back bit is the half of a front bit; 'e' is a terminal symbol: all the symbols after it are ignored. If the number of dimensions of the state space is 2, the first rule in Fig. 3 is decoded as Fig. 4. The working region of a rule is able to overlap with other rules (see Fig. 2). It is the key of dealing with dynamic environment by using proposed system. The last string of the gene determines whether rules participate in parameter vector estimation Eq. (2) or not. The length of the string is equal to the maximum number of rules. '0' means the rule is not available and '1' means available. The string enables the proposed system to vary the number of rules. The gene represents an individual (or agent); it evolves by GA. Each bit of the condition part is inverted or alterative to/from 'e' by mutation.

State space of the environment

B₁ : 110

Working region
of the rule

B₁ : 0100
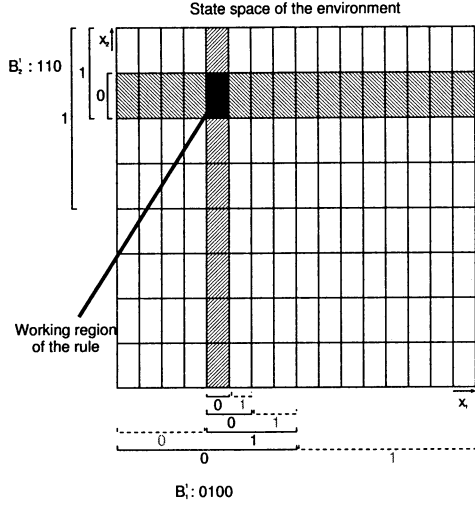
Figure 4: A decoded region

3.4 Learning of rules

We use the SGA to renew the learning parameters. This RL method has low calculation cost; it can deal with various problems since it is not restricted to MDP environment. The general form of SGA algorithm [3] is shown in Fig. 5. Learning parameters of the proposed system (there are correspond to w in Fig. 5) are parameter vector p^k in Eq. (1) and w_1^k, w_2^k in Eq. (3).

4 Experiment

4.1 Experimental method

In this section, we verify the efficacy of the proposed method by mobile robot control under dynamic environment by using Khepera simulator version 2.0 [4]. The dynamic environment is shown in Fig. 6. Striped blocks placed around the center are gates. These two gates are opened initially; they iterate opening and closing in the cycle of 400 steps at the same time. The numbers in Fig. 6 represent (X coordinate, Y coordinate) of the corner of passage and the unit is [mm]. The aim of trials is arrival on upper-right goal. The Khepera robot is a mobile robot that has two driven wheels and 8 sensors.

The RL system observes the value of X coordinate of the robot, the value of Y coordinate, and the value of the direction of the robot (angle). Hence, the number of dimension of the environment's state vector is

1. Observe x_t in the environment.
2. Execute action a_t with probability $\pi(a_t, w, x_t)$.
3. Receive the immediate reward r_t.
4. Calculate $e_i(t)$ and $D_i(t)$ as

$$e_i(t) = \frac{\partial}{\partial w_i} \ln\left(\pi(a_t, w, x_t)\right)$$

$$D_i(t) = e_i(t) + \gamma D_i(t-1)$$

where γ $(0 \leq \gamma < 1)$ denotes the discount factor, and w_i is the ith component of w.

5. Calculate $\Delta w_i(t)$ as

$$\Delta w_i(t) = (r_t - b)D_i(t),$$

wher b denotes the reinforcement baseline.

6. Policy Improvement: update w as

$$\Delta w(t) = (\Delta w_1(t), \Delta w_2(t), \cdots, \Delta w_i(t), \cdots),$$
$$w \leftarrow w + \alpha(1 - \gamma)\Delta w(t),$$

where α is a nonnegative learning rate factor.

7. Move to the time step $t + 1$ and go to step 1.

Figure 5: General form of SGA algorithm

three. The RL system selects and executes one action from three actions of left turn, forward movement and right turn based on 3-dimensional parameter vector p. A reward the RL system receives is $+1$ if the robot arrives at the goal, -1 if the robot does not arrive within 1000 steps. And, if the robot tries to move forward in the state that an obstacle is ahead, the reward -1 is given and the robot turn to the direction to avoid the obstacle. Thus, even though the robot approaches an obstacle, punishment is not given if it does not try to move forward. Hence, the RL system can learn action in the state that the obstacle is approached (e.g., it runs along obstacles).

In this experiment, the length of each crisp set is 4[bits], the string determines

the maximum number of rules is 4000, and genes are initialized randomly. The fitness used in GA is calculated by:

$$\text{fitness} = \xi_1 \, n_{\text{goal}} - \xi_2 \, n_{\text{collision}} - \xi_3 \, n_{\text{rule}}, \quad (5)$$

where $n_{\text{goal}}, n_{\text{collision}}$ are the number of successful trials and colligions in the past 100 trials respectively, n_{rule} is the number of rules, $\xi_1 \sim \xi_3$ are constants and $\xi_1 = 1, \xi_2 = 0.001, \xi_3 = 0.001$. Learning parameters p^k are initialized to $p^k = 0$, and w_1^k, w_2^k are initialized randomly.

We compare the proposed system with conventional look-up table SGA system [3] that has non-time-

Figure 6: Environment

Figure 7: Experimental result (proposed)
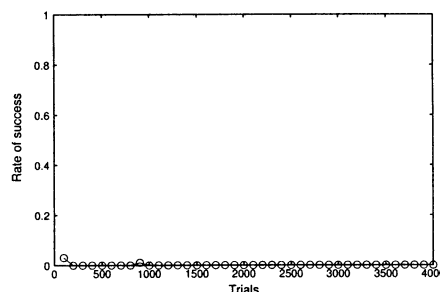
Figure 8: Experimental result (conventional)

varying parameters. The system partition continuous state space into discrete cells. In this experiment, the size of the table is 8^3.

4.2 Experimental result

Fig. 7 and Fig. 8 are results of the proposed and conventional systems respectively (4000 trials in the conventional system correspond to 400 generations). The conventional system cannot learn the action that arrives at the goal since the environment is dynamic: suitable actions are different when gates are opened and closed, and the agent cannot observe the variation of the environment. However, the proposed system is learned since different rules are used when gates are opened and closed.

5 Summary

We proposed an evolutionary and time-varying reinforcement learning system. The proposed system can deal with a dynamic environment using time-varying parameters even if its variation cannot be observed; the division of state space is acquired evolutionarily by GA; one does not have to design the rules constructing an agent in advance. In this paper, we showed the efficacy of the proposed system in unobservable dynamic environment by the control of mobile robot. Future subject is the analysis of the characteristic of the proposed system.

References

[1] A.G. Barto, R.S. Sutton, C.W. Anderson (1983), Neuronlike adaptive elements that can solve difficult learning control problems. IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-13, No. 5, pp. 834-846

[2] H. Kimura, M. Yamamura, S. Kobayashi (1996), Reinforcement Learning in Partially Observable Markov Decision Processes: A Stochastic Gradient Method (In Japanese). Journal of Japanese Society for Artificial Intelligence, Vol. 11, No. 5, pp. 761-768

[3] Kimura H, Yamamura M, Kobayashi S (1997), Reinforcement Learning in POMDPs with Function Approximation. Proceedings of the 14th International Conference on Machine Learning, Vol. 11, No. 5, pp. 152-160

[4] http://diwww.epfl.ch/lami/team/michel/khep-sim/index.html

Application of Direct-Vision-Based Reinforcement Learning to a Real Mobile Robot with a CCD camera

Masaru IIDA   Masanori SUGISAKA   and Katsunari SHIBATA

Dept. of Electrical and Electronic Engineering, Oita University.
700 Dannoharu Oita 870-1192 Japan. mail: iida,shibata@cc.oita-u.ac.jp

Abstract

In this paper, it was confirmed that a real mobile robot with a CCD camera could learn appropriate actions to reach a target by Direct-Vision-Based reinforcement learning (RL). The learning was done online without any pre-process for the task and the environment, and the system consists of one layered neural network. In Direct-Vision-Based RL, row visual sensory signals are put into the layered neural network directly, and the network is trained by Back Propagation using the training signal generated based on reinforcement learning. Although it was afraid that the neural network could not learn appropriate actions sufficiently because of the large number of the input signals, it is shown that the robot could obtain appropriate reaching actions to a target object through the learning from scratch without any advance pre-process in "going to a target" task.

1   Introduction

Reinforcement learning is an attractive method as an autonomous learning for autonomous robots, and has been utilized to obtain the appropriate mapping from state space to action space. By combining the reinforcement learning and a neural network, continuous states and actions can be dealt with because non-linear functions with continuous input and output values can be approximated by the neural network. This combination has been applied to non-linear control tasks[1][2] and games[3].

Among many kinds of sensors for a robot, a visual sensor has a lot of sensory cells, and gives huge pieces of information about the environment to the robot. Our human also depends deeply on visual information to know the environment state. Asada et al. applied reinforcement learning to real soccer robots with a visual sensor[4]. In this case, the state space was divided into some discrete states by pre-processing the visual sensory signals, and the robot learned appropriate actions for each state by Q-learning.

On the other hand, in order to realize intelligence in a robot, the authors think it important that the knowledge for the task is not given to the robot by humans, and the robot obtains the function to achieve the task by itself. The authors have proposed Direct-Vision-Based RL based on this idea. In the learning, the whole process from sensors to motors are computed by a layered neural network. Raw visual sensory signals are put into a layered neural network directly, and the neural network is trained by Back Propagation using the training signal that is generated autonomously inside the robot based on reinforcement learning.

By this learning, not only the motion planning, but also a series of processes from sensors to motors including recognition, can be learned synthetically. It is reported that when a robot learns actions to reach a target, spatial information is represented adaptively on the hidden layer after learning. Moreover, it was examined that the learning is faster and more stable than that when pre-processed information is utilized as inputs on some simulations[5]. The effectiveness of Direct-Vision-Based RL was confirmed by a real mobile robot with a simple visual sensor that had 64×1=64 sensor cells[6].

In this paper, the small CCD camera is introduced to get more visual information, and it is verified that in spite of the many input signals, the real mobile robot with the CCD camera(64×24=1536) can learn appropriate motions from scratch without any advance pre-processing in "going to a target" task.

2   Direct-Vision-Based RL

Fig.1 shows the concept of Direct-Vision-Based RL. Here, actor-critic architecture is employed, and actor(motion command generator) and critic(state evaluator) are composed of one layered neural network as the simulations in [5]. That means that the hidden layer is used for both actor and critic. TD(Temporal Difference) is applied for the critic learning. TD error is defined as

$$\hat{r}_t = r_t + \gamma P_t - P_{t-1}, \qquad (1)$$

where γ : a discount factor, r_t : a reward, P_t : the critic output. The critic output at the previous time P_{t-1} is trained by the training signal as

$$P_{s,t-1} = P_{t-1} + \hat{r}_t = r_t + \gamma P_t, \qquad (2)$$

where $P_{s,t-1}$ is the training signal for the critic output. On the other hand, the motion command of the robot
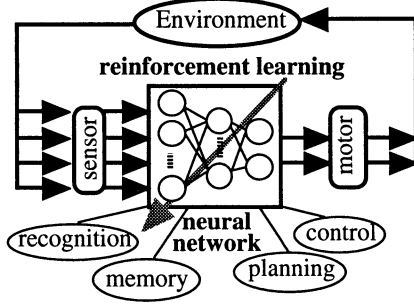
Figure 1: The concept of Direct-Vision-Based Reinforcement Learning.

(a)The picture of the Khepera with a CCD camera.

(b)The visual field of the CCD camera.

Figure 2: A picture of Khepera with CCD camera.

is the sum of the actor output vector a_t and random number vector rnd_{t-1} as trial and error factors. The actor output vector a_{t-1} is trained by the training signal as

$$a_{s,t-1} = a_{t-1} + \hat{r}_t \cdot rnd_{t-1}. \tag{3}$$

The neural network is trained by Back Propagation according to Eq(2) and (3). By this learning, the motion commands are trained to gain more critic output.

3 Application to a real mobile robot

3.1 Experimental system and environment

Fig.2(a),(b) show the robot with a CCD camera (Khepera and CK-200,KEYENCE) used in this experiment. The specifications of Khepera and CCD camera are as follows.

Height : **55mm**
Diameter : **33mm**
Interface with PC : **RS232C(serial poat)**
Transmission rate : **38400 bps**
Number of pixels : **250,000**
Visual field : **114 degree**

The visual field was extended to 114 degree by using a wide angle lens. The video signals of the CCD camera were sent to a PC by NTSC, and the video signals were used as input signals of the neural network.

Fig.3 shows experimental environment. The action area has 70×70cm which is surrounded by a height of 10cm white paper wall, and a fluorescent light is set to keep enough light intensity. The target in the task stands 8cm tall with a diameter of 2.5cm which is wrapped black paper around.

3.2 Coping with a time delay

When Direct-Vision-Based RL is applied to the real robot, a time delay should be considered, while it dose not have to be considered in simulations. The PC receives infra-red sensory signals through RS232C serial port and the video signals through a capture board

Figure 3: Experimental environment.

Figure 4: Timing chart in the real robot.

from CCD camera. The necessary time to execute each process is approximately as follows.

Transmission of video signals : **55msec**
Transmission of infra-red signals : **20msec**
Transmission of motion commands : **10msec**
Computation of neural network(forward) : **20msec**
Computation of neural network(BP) : **40msec**

Fig.4 shows the timing chart of the system events in this experiment. Considering the measurement interval of the each process, sampling time is set to be 310msec. The next motion command should be transmitted just after the capture. However, if the motion commands are sent to the robot just after the computation of the neural network, the robot continues to move with the previous motion commands during the transmission of the video signals and computation

of neural network after capturing the image. As the result, the random number that is added to the actor output is not evaluated exactly. Since the random number is independent with each other, the time delay is expected not to influence the learning so much in the long time scale.

3.3 Discrete actions

Since the action command of Khepera should be an integer, the continuous motion value is discretized into an integer *speed* as

$$speed_t = (int)8 \cdot (\mathbf{a}_t + \mathbf{rnd}_t), \qquad (4)$$

$$if(speed_t \leq -3) \ speed_t = -3$$

$$if(speed_t \geq 3) \ speed_t = 3$$

$$-3 \leq speed_t \leq 3, \ -0.5 \leq \mathbf{a}_t \leq 0.5,$$
$$-0.2 \leq \mathbf{rnd}_t \leq 0.2.$$

where *speed* : the motion signal for the robot.

4 Experiment

4.1 Task

In this paper, the task that the real mobile robot with a CCD camera reaches a target, is employed. At first, the color image with 320×240 pixels is converted to a 8bit(256) gray scale image. Moreover, the number of pixels is reduced to 64×48 by taking an average of 5×5=25 pixels, and the lower half of the image (64×24=1536) was used for the learning. Here, 3-layered neural network with 1536 input units, 50 hidden units, and 3 output units was used. One of the outputs is for critic, and the other two are for actor. Before learning, the input-hidden connection weights are small random numbers, and all the hidden-output connection weights are 0.0. The output function of each hidden or output neuron is sigmoid function whose output range is from -0.5 to 0.5. When the two infra-red sensors at the front of the robot takes 1023 that is the maximum value, the critic output is trained to be 0.9 as a reward. When the robot loses the target, it is trained to be 0.1 as a penalty. Otherwise the critic is trained according to Eq(2) at every time step with $r = 0$. Each trial is stopped at 100 time step even if the robot cannot reach the target object. As the actual critic value, 0.5 is added to the critic output of the neural network. The discount factor γ is 0.98.

Next, it is explained how to determine the initial position of the robot in learning. After the transmission of the visual sensory signals, the first row of them is binalized with the boundary value of 60, and *width* of the target is defined as the number of pixels of the dark area in the robot's view. *Center pixel* is defined as the central pixel number of the dark area. As an initial position, *width* and *center pixel* are chosen randomly within the range where the whole object image can be caught. Then, the robot moves to the initial
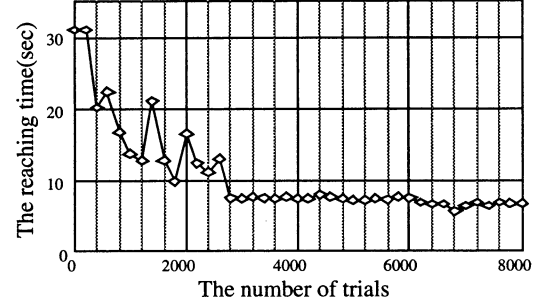
Figure 5: The learning curve.
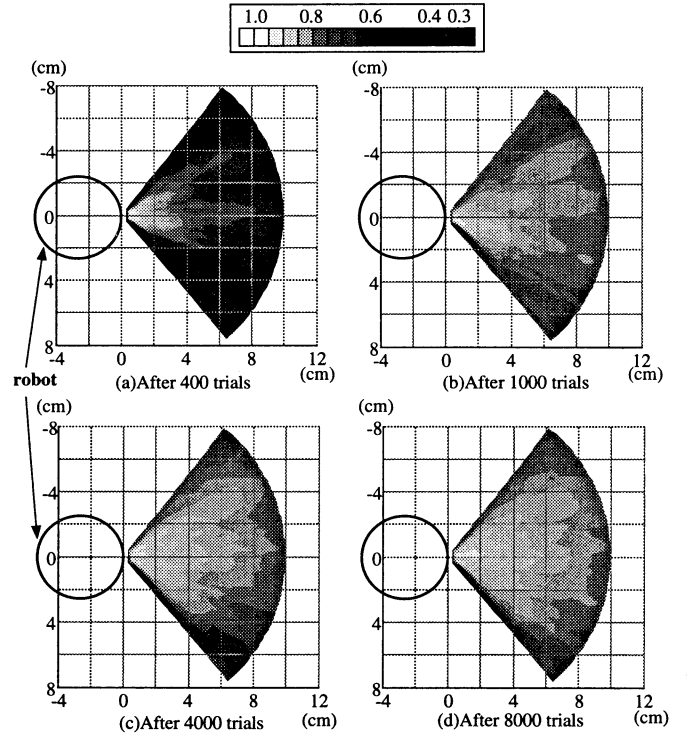
Figure 6: Distribution change of the critic output.

position by itself. At the beginning of learning, since the robot moves only randomly, the robot is located close to the target. As the learning progresses, the initial robot location range becomes wider gradually.

4.2 Learning result

Fig.5 shows the learning curve. The vertical axis indicates the average time(sec) until reaching the target over five target locations, those are (*width,center pixel*) = (6,8) (6,20) (6,32) (6,44) (6,56). The horizontal axis indicates the number of trials. Each trial was stopped after 31(sec) from start even if the robot could not reach the target object. It is seen that reaching time is reduced gradually as the learning progresses.

Fig.6 shows the state evaluation(critic) value after learning. This figure is drawn by computing the critic outputs for 800 sample sets of visual sensory signals.

(a)Disposed target for the test on the field.

(b)The loci of the robot on the field after 8000 trials.
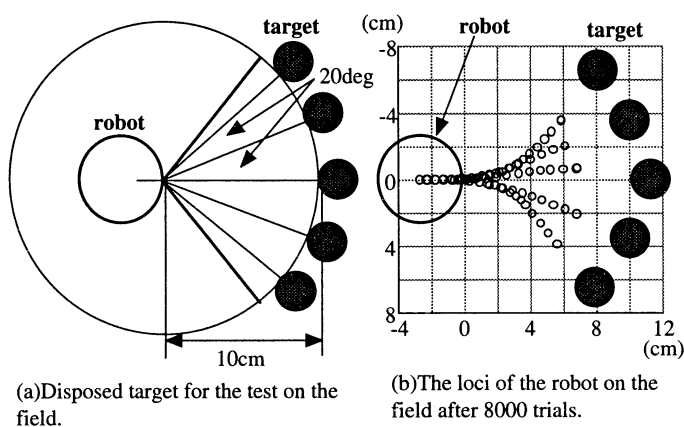
Figure 7: The loci of robot on the field after 8000 trials of learning.

As the learning progresses, the slope of the critic values is formed gradually at first, and then becomes gentle. That is because the necessary time steps for reaching the target became small, while the discount factor is always the same.

Fig.7 shows the loci of the robot on the absolute coordinates after 8000 trials. The robot approaches the target object while rotating. When the robot catches the target object on the center of the robot's view, the robot goes straight to the tartget.

Fig.8 shows a time series of photos to show that the robot reaches the target object without missing after 8000 trials.

5 Conclusion

Direct-Vision-Based RL was applied to the real robot with a CCD camera. Even with a large number of the visual input signals (64×24=1536), it was shown that the robot could obtain appropriate reaching actions to the target through the learning from scratch without any advance pre-processing.

In the future, it is going to be verified that the hidden layer obtains spatial information through the learning from camera image.

Acknowledgements

References

[1] Anderson, C. W. (1989) "Learning to Control an Inverted Pendulum Using Neural Networks", *IEEE Control System Magazine*, Vol. 9, pp.31-37
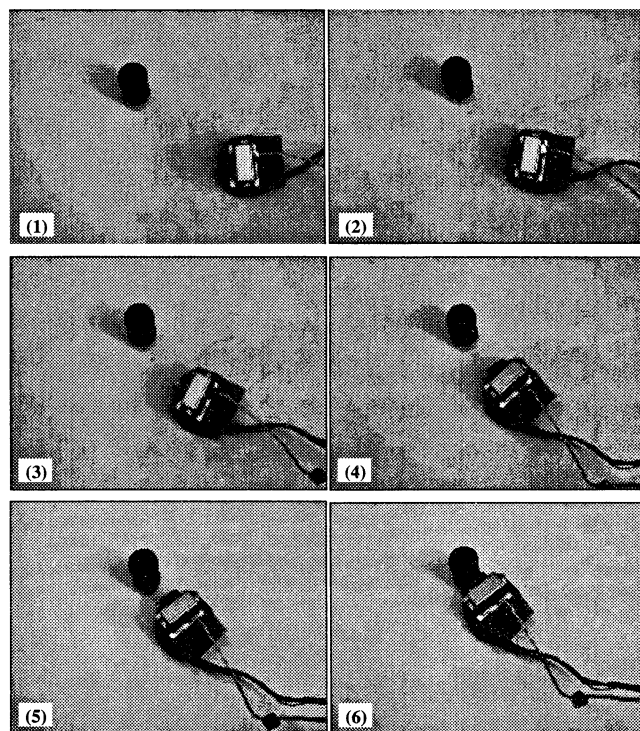
Figure 8: The robot succeeded in reaching a target object after 8000 trials of learning.

[2] Morimoto, J., Doya, K. (2001) "Acquisition of Stand-up Behavior by a Real Robot using Hierarchical Reinforcement learning", *Robotics and Autonomous Systems*, 36, pp.37-51

[3] Tesauro, G. J. (1992) "Practical Issues in temporal difference learning", *Machine Learning*, 8, pp.257-277

[4] Asada, M., Noda, S., Tawaratsumida, S. and Hosoda, K. (1996) "Purposive Behavior Acquisition for a Real Robot by Vision-Based Reinforcement Learning", *Machine Learning*, Vol.24, pp.279-303

[5] Shibata, K., Ito, K. and Okabe, Y.(1998) "Direct-Vision-Based Reinforcement learning "Going to a Target" Task with an Obstacle and with a Variety of Target Sizes", *Proc. of NEURAP'98*, pp.95-102

[6] Iida, M., Sugisaka, M. and Shibata, K.(2002) "Application of Direct-Vision-Based Reinforcement Learning to a Real Mobile Robot", *Proc. of ICONIP(Int'l Conf. on Neural Information Processing)'02*, CD-ROM Proc., CR1876.pdf, 2002.11

[7] Barto, A. G., Sutton, R. S. and Anderson, C. W. (1983) "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Probrems", *IEEE Trans. SMC-13*, pp.835-846

Body Growth Effect on Reinforcement Learning
in a Simple Standing-up Task

Daiki KIYOSUKE, Masanori SUGISAKA and Katsunari SHIBATA, Oita University

Abstract : The authors believe that the body growth accelerates the learning of actions, such as standing up in higher forms of lives. While, the slow learning is a serious problem in reinforcement learning. In this paper, it was verified whether the body growth accelerates the learning in a simple standing-up task. It was shown that even though the pendulum could not learn to stand-up when its arm was long, it could learn when the arm become longer from short state during the learning.

1. Introduction

A man is born in an immature state, and then grows up gradually. Why does man grow up? As one answer for this question, the authors think that growth has an effect to improve the learning speed drastically. A man starts walking around 1 year old. At this time, even though there are differences among individuals, the height is below a half and the weight is below 1/5 of a grown-up. If a man is going to learn with the grown-up body, it can be imagined easily that it is difficult to learn standing-up based on trial and error. It is also thought that when a man with a grown-up body falls down to the ground from the standing state, the damage is seriously larger than a child. Accordingly it can be thought that the learning with growing-up accelerates the learning speed and also makes a role to avoid the dangers.

In recent years, reinforcement learning, which is an autonomous learning method getting a hint from the learning of living things, has been focused. An advantage of this learning is that autonomous and flexible action learning can be realized based on trials and errors. However, since the learning contains the element of search, there is an important problem that it takes a long time to learn. For this problem, it has been proposed to raise the difficulty gradually with changing the environment of a learning task[1]. In that paper, the task to balance on a bicycle by the action of pedaling was employed. The difficulty of the task was increased by moving the auxiliary wheel upwards gradually. As a result, acceleration of the learning was observed. However, the effect of body growth has not been mentioned yet.

On the other hand, Barto et al. pointed out that when the task of hand-stand pendulum was learned by rein-

forcement learning, it is more difficult to balance with a short arm than with a long arm[2]. The walking mentioned above is also thought of an extension of the inverted pendulum. This seems to suggest that a grown-up man can learn walking easier than a child, and to contradict to the author's hypothesis.

In this paper, body growth is introduced into reinforcement leaning, and the body growth effect is observed by the simulation of a simple standing-up task. Furthermore, it is shown that Bartos' result does not deny the body growth effect based on some consideration.

2. Reinforcement learning

In this paper, actor-critic architecture is employed for reinforcement learning. Critic(state evaluation) is learned by TD (Temporal Difference) learning. Using the present critic output P_t, TD error is defined as

$$\hat{r}_t = r_t + \gamma P_t - P_{t-1},\qquad(1)$$

where γ : a discount factor, r_t : a reward. To decrease $\hat{r}_t$, the previous critic output P_{t-1} is updated by the following equation.

$$P_{t-1} = P_{t-1} + \alpha\hat{r}_t,\qquad(2)$$

where α : a leaning constant for the critic. The actor's output $\mathbf{a}_t$ added by a random number $\mathbf{rnd}_t$ is used as an actual action. Using TD error, the previous action $\mathbf{a}_{t-1}$ is updated by the following equation.

$$\mathbf{a}_{t-1} = \mathbf{a}_{t-1} + \beta\mathbf{rnd}_{t-1}\hat{r}_t,\qquad(3)$$

where β : a leaning constant for the actor. If $\hat{r}_t$ is large, $\mathbf{a}_{t-1}$ is reinforced to $\mathbf{a}_{t-1} + \mathbf{rnd}_{t-1}$ to get more gain of the critic output.

3. Task

Fig.1 explains the task employed in this paper. The task is that a simple 1-link pendulum stands up from the state of lying down. Only when the pendulum stands vertically within a limited angular velocity, a reward is given, and the trial finishes.
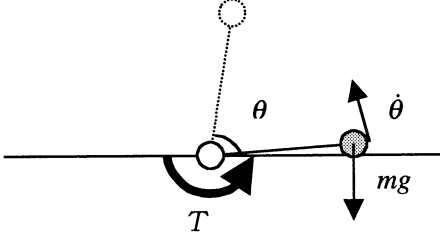
Fig.1 A simple standing-up task

When the pendulum falls down to the floor, the angular velocity is set to be 0.0 and the trial continues. The state space consists of 45*101 states. The joint angular velocity is divided into 101 and the joint angle divided into 45. The evaluation value P and output joint torque T for each state are held as a table, and are updated by reinforcement learning. 10,000 seconds is defined as 1 cycle. The learning continues for 500,000 seconds. When 1cycle passes or pendulum arrives at the goal state, the pendulum is returned to the initial state $(\theta, \dot{\theta})=(0,0)$. The reward 1 is given when the angle from the vertical axis is less than 8 (rad) and the angular velocity is less than 0.84(rad/sec). No friction is assumed.

4. Body Growth

As the body growth, the length of pendulum is changed linearly from 0.3m to 1.0m during 400,000 seconds. The mass of the pendulum weight and the maximum output torque is determined as follows. The specific gravity of the pendulum weight is assumed to be constant and the shape is assumed to be the same. Then the mass is calculated as

$$m = l^3, \qquad (4)$$

where m: the mass of the pendulum weight, l: the length of the pendulum. Moreover, the maximum output torque is assumed to be proportional to the torque that is necessary to keep the pendulum horizontally, and is calculated as

$$Tmax = T\_MAX \times ml = T\_MAX \times l^4, \qquad (5)$$

where $Tmax$: the maximum output torque, T_MAX: a constant. Here, since T_MAX is set to be 40, the range of the torque is changed from $-0.324{\sim}0.324$(Nm) to $-40{\sim}40$(Nm). The actual torque as the action is calculated as

$$T = (a_{x,t} + rnd_t^3)T\max, \qquad (6)$$

where rnd_t is an uniform random number from -1 to 1. If the $a_{x,t} + rnd_t^3$ is smaller than -1 or bigger than 1, it is set to be -1 or 1. In other words, the ratio of the output torque in the maximum torque is learned.

5. Simulation

For comparison, the learning was done in three cases. In the first case, the length of the pendulum was fixed to the maximum (1.0m), while in the second, it was fixed to the minimum (0.3m). In the last case, the pendulum grew up. Here, the other conditions are the same. The learning constant for the critic α was 0.3, that for the actor β was 4.0, and the discount factor γ was 0.99.

When the arm length was 1.0m, the number of successes is too few for the pendulum to learn standing up. Fig.2 shows the number of average steps until it succeeded to stand up. Although the learning speed in the case of growing pendulum is a little slower than the case when the arm length was fixed to 0.3m, the number of average steps converged in both cases. The converged steps for the 0.3m arm length is slightly smaller.

Fig.3,4,6 and 7 shows the distribution of the critic value and the ratio of the output torque in the maximum torque in the cases of the 0.3m arm length and the growing arm respectively. The goal is indicated by a rectangle at the center and the trajectory is indicated by gray squares in each figure. In Fig.3 and 4, it can be seen that the slope of the critic output is formed from initial state toward the goal. The maximum angular velocity in the area of large critic output is larger in Fig.3 than in Fig.6. That is because the maximum acceleration is lager with a short arm than with a long arm by the relation between the maximum torque and the rotary inatia.
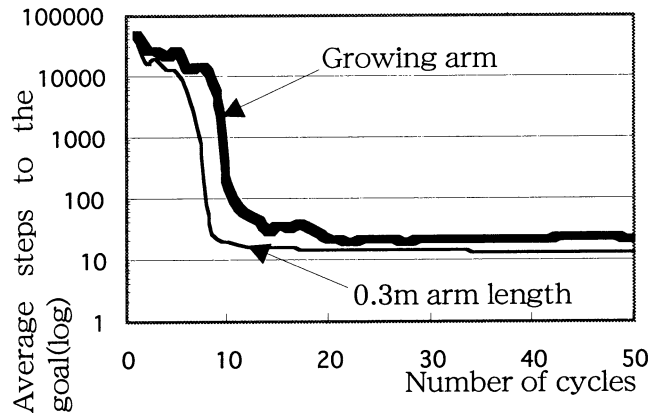
Fig.2 Comparison of the learning curve
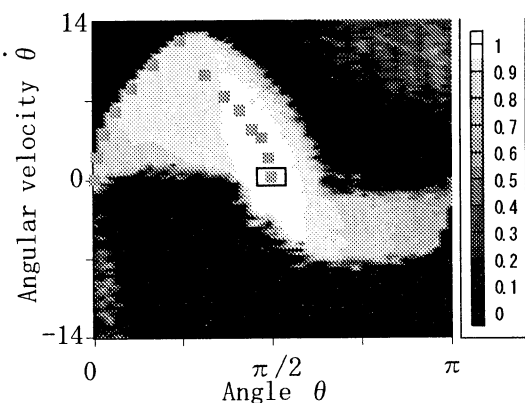
Fig. 3 Critic output as a function of the pendulum state (l=0.3m)
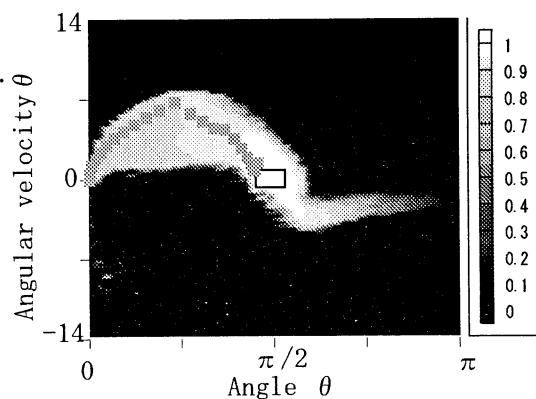
Fig. 6 Critic output as a function of the pendulum state (l:0.3m→1.0m)
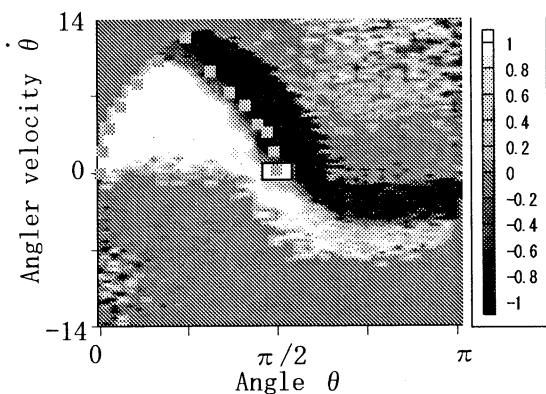
Fig. 4 Actor output (action) as a function of the pendulum state (l=0.3m)
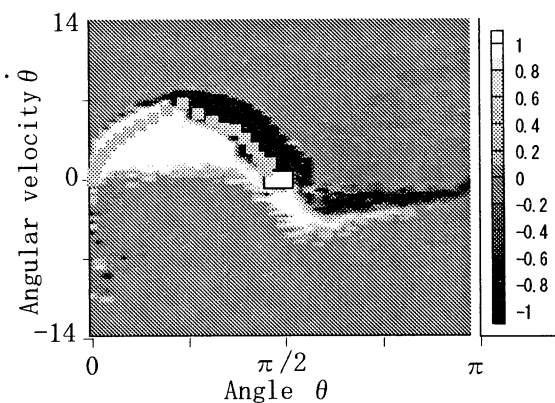
Fig. 7 Actor output (action) as a function of the pendulum state (l:0.3m→1.0m)
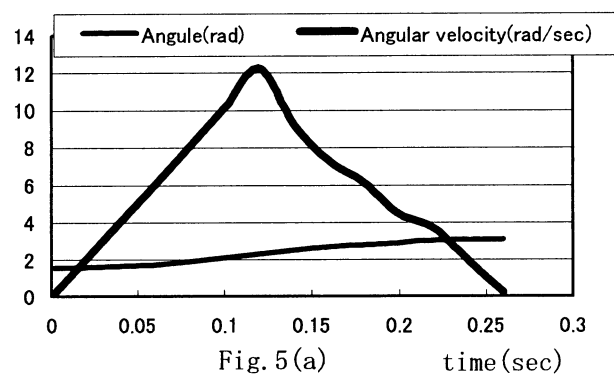
Fig. 5 (a)
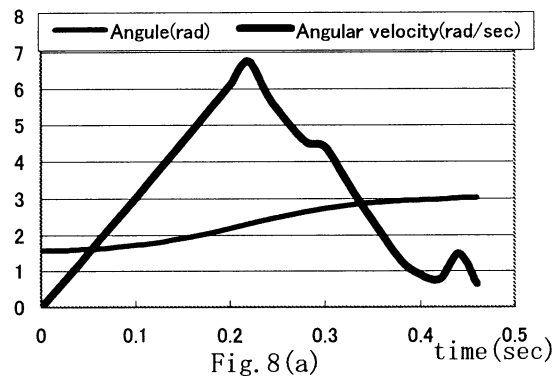
Fig. 8 (a)
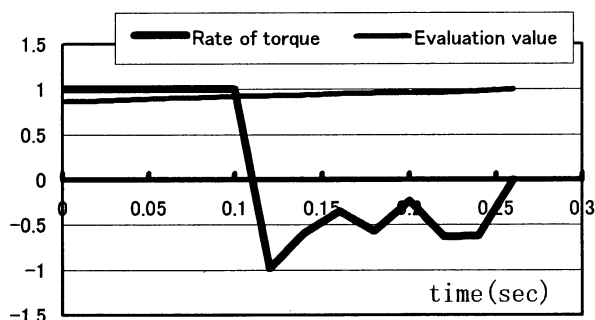
Fig. 5 (b)
Fig. 5 Change of the variables in one trial after learning (l=0.3m)
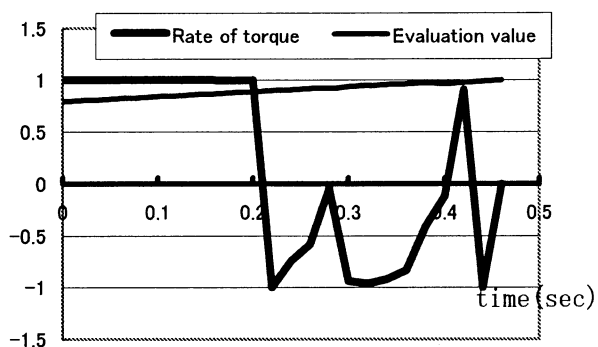
Fig. 8 (b)
Fig. 8 Change of the variables in one trial after learning (l:0.3m→1.0m)

-92-

As shown in Fig.4 and 7, it is known that the pendulum applies the break by the negative torque to stand up within the limited speed when the angle is larger than $\pi/4$. In the case of 1.0m arm length, the learning of the action didn't progress because the critic output becomes large only around the goal and the slope of the critic from the initial state was not formed. Fig.5 and 8 shows that the change of the angle and the angular velocity to the goal in Fig.(a) and the change of the ratio of the output torque in the maximum torque and the critic output in Fig.(b) for the cases of 0.3m arm length and growing arm respectively. It is known that the pendulum accelerates at first and then decelerates to stand up with almost 0 angular velocity.

From these leaning results, by growing the arm from 0.3m during learning, the pendulum becomes to stand up even with 1.0m arm length. That is because the number of successful trial becomes larger when the arm is short.

6. Discussion

In this simulation, the result is deeply depending on the parameters. Especially the way to determine the maximum torque such as Eq.(5) gives strong influence to the result. At equation (5), it was presupposed to be proportional to l^4. That is based on the setup that it is proportional to the torque to support the arm horizontally. In this case, since the rotary inatia is proportional to $ml^2 = l^5$, the ratio of the maximum acceleration becomes $1:l$. Accordingly, if the influence of the gravity is neglected, the minimum time until standing-up becomes $\sqrt{l}:1$. This is the same as the ratio of the natural frequency exactly. In the previous simulations, since $l=0.3$ or 1m, the ratio of one cycle is 1.83.

From Fig.5 and 8, the ratio of the time to reach the goal is just close to this value. However, if the maximum torque is adjusted so as that the maximum angular acceleration becomes the same, the maximum torque should be proportional to l^5. In this case, if the influence of gravity is neglected, the time until the pendulum standing-up becomes equal. If the influence of gravity is considered, conversely, the pendulum with longer arm length can stand-up more quickly. That does not seem natural.

A man generates his joint torque by his muscles. It is assumed simply that since a cross-section area of each muscle is proportional to l^2, muscle tension is proportional to l^2. Moreover, since the moment arm is proportional to l, the torque is proportional to l^3. Though

this value should be measured actually, it is clear that l^5 is much against the scaling law, and l^4 is more appropriate.

To the next, the relation between the minimum time to stand up and the learning are considered. If the time to stand up is long, the torque to the same direction must continue to be outputted. In reinforcement learning, since the learning is based on trial and error, if the necessary time is long, the probability of standing-up becomes small. If the control interval is long, this does not become a big problem. In the above simulation, the pendulum can learn stand up even with 1.0m arm length when the interval is doubled. However, the interval is required to be short to realize fine control.

At last, Bartos' result is considered. In their experiment, the purpose is that the pendulum that is starting initially continues standing. Therefore, it is easy for the longer pendulum to be controlled since the time until it falls down is longer to a contrary. However, standing-up has to be learned before balancing, and standing-up is difficult to learn than balancing because it returns to the initial state when it falls down. Moreover, in the case of Bartos' simulation, it is unnatural that the applied power is constant not depending on the length.

7. Conclusion

In this paper, it was shown that the body growth has the effect to accelerate reinforcement learning drastically in a simple standing up task. The speed is deeply depending on the maximum torque, but the validity of the set up in this paper was shown. In the future, the way of growth will be considered and a task with many degrees of freedom will be employed.

Bibliography

[1] Jette Randlov,(2000) Shaping in Reinforcement Learning by Changing the Physics of the Problem Proc.of ICML, p.p.775-782

[2] A.G.Barto, et al.(1985) TRAINING AND TRACKING IN ROBOTICS, In Proc.of IJCAI,p.p.670-672

Distributed Mobile Robotic Systems Applied with Behavior Models of a Fish School

Tatsuro Shinchi*, Masayoshi Tabuse**, Tetsuro Kitazoe**, Akinobu Todaka**, Takahiro Horita**

* Faculty of Education and Culture, ** Faculty of Engineering, Miyazaki University

Gakuen Kibanadai Nishi 1-1, Miyazaki City, Japan, 889-2192

shin@edc.miyazaki-u.ac.jp

Abstract

This paper presents behavior models of a mobile robot, which employ the character of fish school movements. Though a fish school does not need the special individual which leads the entire, an autonomous movement emerges from interacting among many-bodies. This study aims to realize autonomous collective movements of mobile robots through competitive and cooperative interaction among robots. We, therefore, improved the behavior model of a fish to adapt to the specification of the mobile robot. The emergence of the safe multi-robots movements through simple interaction can be said the realization of distributed robotics systems.

As a result of simulations combined with genetic algorithms, we obtained a beautiful collective movement of multi-robots, with high techniques to avoid collisions, speed up, slow down, change lanes or overtake a slow robot.

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 free style of driving; every autonomous robot on a road have its own way of driving and can choose its own speed. Some transportation systems have succeeded in realizing collective movement of vehicles. Those movements, however, are quite rigid, individuals run just in file. Namely, vehicles run by only keeping a good distance from the robot ahead, or by only following markers embed in the road.

In the present study, we use a fish school algorithm for autonomous robot movements[1]. Fishes have a beautiful smooth movement in a school without making any collision at all. It is well known that a fish decides its motion by getting just from the neighboring fishes without knowing anything of the whole

school[2]. Since the fish school algorithms are well known[3], we improve behavior models of a fish to adapt to the specifications of Khepera. In this study, we introduce the genetic algorithms to train behavior rules for the smooth movement without accident. Optimization with genetic algorithms, the frequency of collision has dramatically reduced. Moreover, it was sometimes observed that some robots had a nice driving skill, overtaking another robot with adjusting the velocity and the direction.

2 Experimental Environment

2.1 Mobile Robot and Simulator

In our experiments, we assume to use a miniature mobile robot Khepera.[4] 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. The difficult problem in the computer simulations is how to simulate the real world with a proper treatment of noise. *Khepera Simulator ver.2.0* written by Olivier Michel (downloadable from the *WWW* [5])is an excellent software by taking account of uncertainty which is supposed to exist in the real world. In this paper, we investigate movements in computer simulations by employing *Khepera Simulator ver.2.0*.

2.2 Conditions in the Simulator

We set up the two kinds of roads in our simulations, one is the straight road and another is the wind road,

as shown in Fig.1. Two roads in Fig.1 are linked each other. Therefore, robots run repeatedly on the looped way of a) and b) in Fig.1. The maximum motor speed of 5 robots is set as 4(faster) and the maximum speed of other 5 robots is set as 2, where 10 robots run on the road together. Each robot frequently changes its direction according to the returned values from the installed 8 sensors.
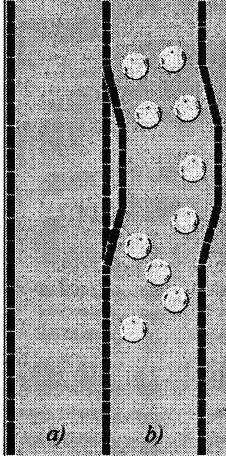
Fig. 1: The road in simulations.

3 Behavior Models of A Robot
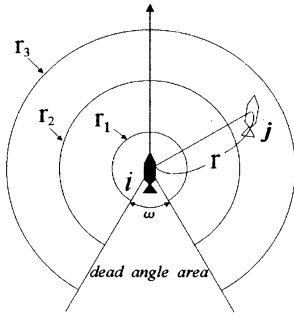
Fig. 2: Ranges for a fish behavior.

Since we apply the behavior model of a fish(Fig.2), sensors equipped on Khepera should be used like as the sense organ of a fish, such as eyes and lateral lines. Therefore, for interactive rules between robots, rear sensor ♯6 and ♯7 are not used. 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 couples of s_0,s_1 and s_4 s_5 are integrated as $0.5 \cdot (s_0 + s_1)$ or $0.5 \cdot (s_4 + s_5)$, respectively.
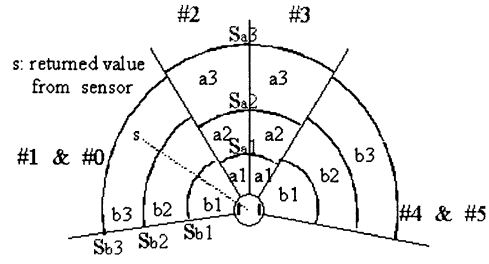
Fig. 3: Discernible Rages of Khepera Sensors

The area which determines the behavior type of robot is divided concentrically with S_1,S_2,and S_3. The most inside boundary is defined with S_1, next boundary is defined with S_2, and the most outside boundary is defined with S_3 shown as Fig.3. Since sensors on Khepera are equipped symmetrically, a) sensors of ♯2 · ♯3 detect the right ahead regions with radius S_{a1}, S_{a2}, S_{a3}, b) sensors of ♯0 · ♯1 and ♯0 · ♯5 detect left or right regions ranged with S_{b1}, S_{b2}, S_{b3}.The distance to an obstacle is measured with the returned value s. Tab.1 and Tab.2 show the behavior type and the concrete behavior, respectively.

area	a) ♯2, ♯3	b) ♯0♯1, ♯4♯5
$s \geq S_1$	avoidance	avoidance
$S_1 > s \geq S_2$	avoidance	parallel
$S_2 > s \geq S_3$	approach	approach
$S_3 > s$	search	search

Tab.1: The types of behavior corresponding to the area where the obstacle is detected. "avoidance" means avoidance of collisions against other robots or guardrails. "parallel" means advance robots side by side or advance parallel to guardrails. "approach" means approach to robots or guardrails. "search" means search in free space without obstacles.

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.2: The robot behavior corresponding to the area where the detected obstacle is. At the behavior type of "quick turn", each wheel of both sides rotates opposite direction each other. At the behavior type of "slow turn", the speed of the one-side wheel increases.

4 Collective Movements of Multi-Robots

4.1 Collective Movements with Manually Set Models

Firstly, we have the simulations with S shown in Tab.3, which are set manually. In our model, it is unique that the order of detecting the area is set freely. In the simulations with Tab.3, the order is $a_1, b_1, a_2, b_2, a_3, b_3$, and outside of a_3 and b_3. When an obstacle is detected in the area, a robot behaves following Tab.1 and Tab.2. .

area	a) ♮2, ♮3	b) ♮1♮4, ♮0♮5
S_1	900	900
S_2	600	600
S_3	300	300

-5mm Tab.3: The value which determine radius in Fig.3.

As the result of simulations, robots run together along the forms of a road. However, a robot often cannot avoid collision as shown in Fig.4. Fig.?? is one of moments, where robots collide with other robots. It is assumed that $S_{1\sim3}$ are not adapted to the situation of the road and Khepera.
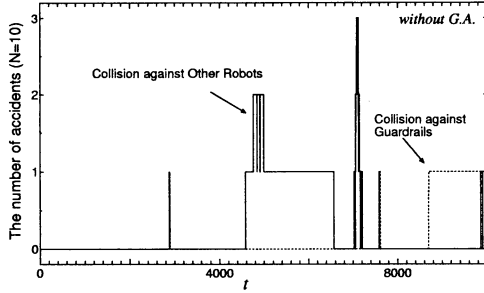
Fig. 4: The frequency of collision against other robots. The number of robots (N) is 10.
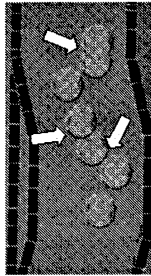
Fig. 5: The collision of robots, where they overlap each other.

4.2 Collective Movements with Models through G.A. Operation

We introduce genetic algorithms to optimize simultaneously $S_{a1} \sim S_{a3}, S_{b1} \sim S_{b3}$ and the order of detecting area. The length of one gene is 12(for $x_1, x_2, \cdots, x_{12}$ which takes values between 0.0 and 1.0. The former 6 chromosomes ($x_1 \sim x_6$) determine $S_{a1} \sim S_{a3}$ and $S_{b1} \sim S_{b3}$. The latter chromosomes ($x_7 \sim x_{12}$) determine the order of detecting each area. The obtained gene will be applied equally to all 10 robots. We introduce 40 genes in one generation. Namely, 40 behavior types of multi-robots are evaluated in one generation. $S_1 \sim S_3$ are obtained as

$$
\begin{aligned}
S_1 &= 1024 \cdot x_{1,4}, \\
S_2 &= S_1 \cdot x_{2,5}, \\
S_3 &= S_2 \cdot x_{3,6}
\end{aligned}
\tag{1}
$$

As for the determination of the order of detecting, the area whose chromosome is higher has a prior claim to be detected. When any obstacle is not find in $a_{1\sim3}$ or $b_{1\sim3}$, a robot goes straight.

The evolutional algorithms to obtain the best genes are as follows.
1) 40 genes at the first generation are randomly generated. Next, let multi-robots run on the road for 5000 simulation steps.
2) After the evaluation of 40 behavior types of multi-robots in one generation, 10 best genes are selected as the offspring. Other 30 offspring are generated from the selected 5 genes with both mutation operations $rate$: 20.0%) and crossover operations ($rate$: 100.0%). Then, let them run for the same period as 1).
3) All 40 types of the collective movements with each gene are evaluated by fitness function. And again the 10 best genes are selected with high scores. Evolutionary computation repeats 100 generations in the process of 1) and 2).

The evaluation function in our experiment is given as

$$
g = EF5/(EF1 \cdot EF2 \cdot EF3 \cdot EF4)
\tag{2}
$$

where EF1 is the average mutual distance among robots, EF2 is the total number of collisions against guardrails, EF3 is the total number of collisions against robots, EF4 is the total of direction angles of robots, and EF5 is the average of traveled distance in y-direction. Consequently, the evaluation function in this study leads highly collective multi-robots movements which travel long with avoiding collision against

robots. In this evaluation function, EF2 and EF3 give the significant effect.

After genetic algorithms above, we obtained two typical parameters are showed in Tab.4 and Tab.5.

$sensor$	a) ♮2, ♮3	b) ♮1♮4, ♮0♮5
S_1	779	488
S_2	215	164
S_3	36	142

Tab.4: The obtained value(I) by G.A. which determine radius in Fig.3. Where the order of detecting is $b_1, a_2, a_1, a_3, b_3, b_2$ and lastly other outside area.

$area$	a) ♮2, ♮3	b) ♮1♮4, ♮0♮5
S_1	786	482
S_2	217	61
S_3	37	53

Tab.5: The obtained value (II) by G.A. which determine radius in Fig.3. Where the order of detecting is $a_1, b_1, a_2, b_3, a_3, b_2$ and lastly other outside area.

The collective movements trained with genetic algorithms operation are quite smooth without any collision. Robots interact to avoid collisions each other and all robots go forward along one-way road. Any robot goes backward with U-turn. Interactions among multi-robots contribute to keep their safe and smooth travels. The difference of behaviors following Tab.4 and Tab.5 are given as Fig.6, which indicates the behavior following Tab.4 is superior in going forward along a road.
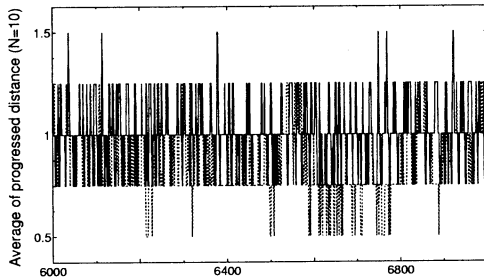
Fig. 6: The average of progressed distance. Where the solid line shows the behavior following Tab.4, the dotted line shows the behavior following Tab.5.

5 Conclusion

We studied free style movement of multi-robots on the Khepera robots simulator. A fish school algorithm is applied to manipulate multi-robots. As the result of the simulation with genetic algorithms operation,

$S_{a1 \sim b3}$ and the order of detecting area are optimized well. Consequently, an autonomous behavior is observed, where robots run well without any collision against guardrails and other robots. Moreover, when there is enough space ahead, the fast robot the slow robot with changing direction. It is assumed, for an autonomous behavior of robots, that the distance regulation between cars is important. Therefore, $S_{a1 \sim b3}$ are critical in determining the character of robot behaviors, which realize feasible distance to the robots and guardrails.

In this study, something special system is not needed to realize autonomous multi-robots movements. The safe and smooth runs are realized solely due to the repetitions of single interactions among robots. That can be said the realization of the distributed robot control system[6]. Therefore, it is assumed that the fish school algorithms is applicable to the distributed multi-vheicle systems.

For further study, assuming more complex and long road, the relation between the number of robots and the effectiveness of multi-robots behavior should be compared, because we need to discuss the swarm intelligent and the suitable number for the road.

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] E.Shaw:The Schooling of Fishes, Scientific American, vol.206, pp.128-138(1962)

[3] Ichiro Aoki:A Simulation Study on the Schooling Mechanism in Fish, Bulletin of the Japanese Society of Scientific Fisheries,48(8), pp.1081-1088(1982)

[4] K-Team SA: Khepera USER MANUAL Version 4.06. (1995).

[5] 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/

[6] Gerado Beni and Jing Wang: Theoretical Problems for the Realization of Distributed Robotic System, Proc. of the 1991 IEEE International Conference on Robotics and Automation(1991)

Navigation Systems for a Wheelchair Based on a Single Camera

Yasuyuki Inoue*, Masayoshi Tabuse*, Yoshinobu Kitamaru*, Tetsuro Kitazoe*, Tatsuro Shinchi**

*Department of Computer Science and Systems Engineering, Faculty of Engineering

**The Center of Educational Research and Practices, Faculty of Education and Culture

Miyazaki University, 1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192, Japan

{yasu@eagle.cs, tabuse@cs, yosinobu@cs, kitazoe@cs, shin@edc}.miyazaki-u.ac.jp

Abstract

In this research, we propose navigation system for an electric wheelchair using a single camera. The navigation system perceives the surrounding environments using a camera, extracts a boundary line of a corridor and a wall and controls a wheelchair. We investigate that a wheelchair controlled by this system runs safely and navigates to a goal.

1 Introduction

Recently, many people have been researching autonomous mobile robots extensively. One purpose of this research is to build the robots that are able to run safely in unknown or dynamically changing environments. In order to perceive the surrounding environments, robots mount many kinds of sensors, e.g., infrared sensors, ultrasonic range sensors and CCD cameras.

On the other hand, as the number of senior people has been increasing, there is a growing demand for more safe and comfortable wheelchairs as mobile aids. There are two research issues with these wheelchairs. One is on autonomous (or semi-autonomous) and safe navigation, such as avoiding obstacles, wall following, going to a goal using various sensors [1,2]. The other is on human interfaces for easy operation [3,4].

In this paper, we propose navigation system for an electric wheelchair using a single camera. In the computer vision, researches of recognition of 3 dimensional information in the environment have been performed. However, generally this approach has large amount of calculation and it is difficult to process 3 dimensional information in real time. Therefore, we consider navigation system based on 2 dimensional picture image obtained by a single camera. The navigation system perceives the surrounding environments using a camera, extracts a boundary line of a corridor and a wall and controls a wheelchair. We investigate that a wheelchair controlled by this system runs safely and navigates to a goal.

2 Outline of Wheelchair system

The wheelchair system in our research is composed of a commercial electric wheelchair (Matsunaga MD-100), a desktop PC (CPU:PentiumII 400MHz, OS:Linux) and a CCD camera as shown in Figure 1. The wheelchair has the maximum speed of 4.5 [km/h]. The CCD camera has a function of pan and tilt, mounted on the front and captures front images. The PC processes picture images of the CCD camera and controls motors of the wheelchair.
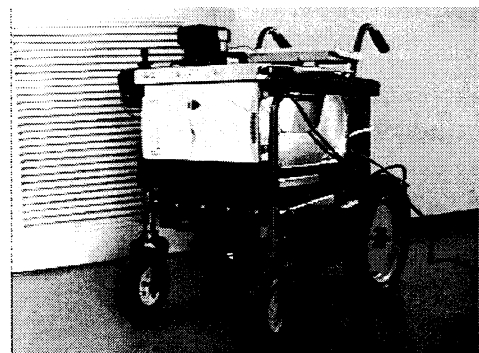
Figure 1: Wheelchair system

3 The Picture Image Processing Methods

Our wheelchair system captures picture images by using a single camera mounted on a wheelchair and navigates a wheelchair based on boundary lines of a passage and a wall extracted from acquired images. The method of picture image processing is as follows.

1. The camera captures a grayscale image.
2. Extracting edges of the image using a gradient filter.
3. Binary imaging by a threshold which is determined by discriminant analysis.
4. Thinning
5. Determining a boundary line of a passage and a wall by Hough transform.

3.1 Discriminant Analysis

In order to obtain binary images from grayscale images, we set

a threshold for grayscale images. However, brightness of images changes dynamically, so that we use discriminant analysis to determine a threshold dynamically due to the brightness.

Intensities of pixels have values between a and b. We separate these intensities into two groups by a threshold t. One has intensities between a and $t-1$ and average intensity $\bar{f}_0$. The other has intensities between t and b and average intensity $\bar{f}_1$. Variance between groups σ_B^2 and variance within groups σ_I^2 are expressed by

$$\sigma_B^2 = \frac{\sum_{k=a}^{t-1} n_k (\bar{f}_0 - \bar{f})^2 + \sum_{k=t}^{b} n_k (\bar{f}_1 - \bar{f})^2}{\sum_{k=a}^{b} n_k}, \quad (1)$$

$$\sigma_I^2 = \frac{\sum_{k=a}^{t-1} n_k (k - \bar{f}_0)^2 + \sum_{k=t}^{b} n_k (k - \bar{f}_1)^2}{\sum_{k=a}^{b} n_k}, \quad (2)$$

respectively, where $\bar{f}$ is average intensity of the image, n_k is the number of pixels with intensity k. Then we calculate $F = \sigma_B^2 / \sigma_I^2$ and determine a threshold t_{max} in which F has the maximum value.

3.2 Hough Transform

To detect a boundary line of a passage and a wall in a image, we utilize the $\theta - \rho$ Hough transform. A straight line through a point (x, y) is denoted by

$$\rho = x \cos \theta + y \sin \theta. \quad (3)$$

So the set of lines passing through each point $p_i(x_i, y_i)$ is represented as a set of sine curves in $\theta - \rho$ space and multiple hits (θ_P, ρ_P) in $\theta - \rho$ space indicate a straight line in the image (see Figure 2).
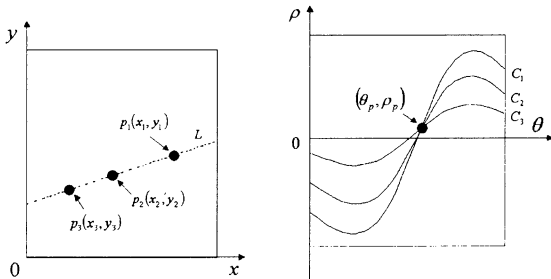
Figure 2: $\theta - \rho$ Hough transform.

4 Navigation Methods

When a mobile robot performs an autonomous run, the most important point is that the robot recognizes its situation and runs safely. In our research, we consider wheelchair navigation based on boundary lines of a passage and a wall, which are extracted from picture images of a single camera in the image processing method mentioned in Sect.3. In order to recognize left- and right-hand lines exactly, we divide the images into left and right image regions as shown in Figure 3. In the left and right image regions, the boundary lines are indicated by Eq.(4) and Eq.(5), respectively.

$$y = a_L x + b_L, \quad (4)$$

$$y = a_R x + b_R. \quad (5)$$
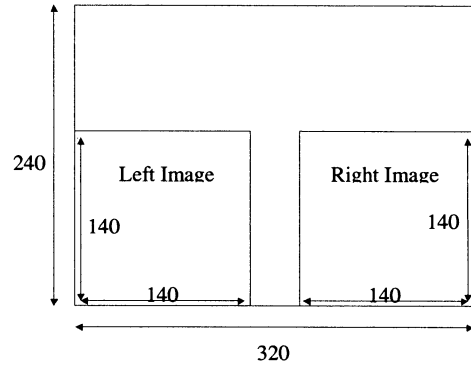
Figure 3: Left and right image regions.

Let us explain navigation methods in the following 3 situations:

1. Facing a corridor.
2. Facing a turning point on one side.
3. Facing a wall.

(1) Facing a corridor

If a wheelchair faces a corridor, both left and right image regions contain a boundary line of a passage and a wall as shown in Figure 4. So a_L and a_R in Eq.(4) and Eq.(5) satisfy the following relations.

$$a_L > 0, \quad a_R < 0. \quad (6)$$

The wheelchair runs keeping two boundary lines in the left and right image regions.
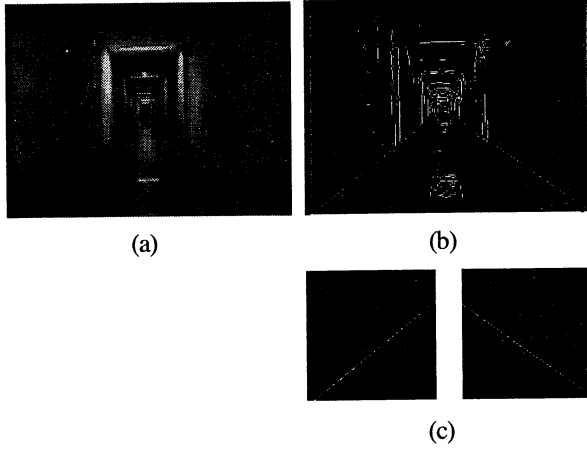
Figure 4: images when the wheelchair faces a corridor. (a) a picture image. (b) a binary image after tinning. (c) left and light images after Hough transform.

(2) Facing a Turning Point on One Side

If a wheelchair faces a turning point on one side, left or right image region contains a boundary line of a corridor and a wall around the turning point as shown in Figure 5. So a_L and a_R in Eq.(4) and Eq.(5) satisfy

$a_L > 0$, $a_R \approx 0$ for a turning point on right-hand side, (7)

$a_L \approx 0$, $a_R < 0$ for a turning point on left-hand side. (8)

Then the system recognizes the turning point and pans the camera to the turning direction to confirm that there is a corridor.
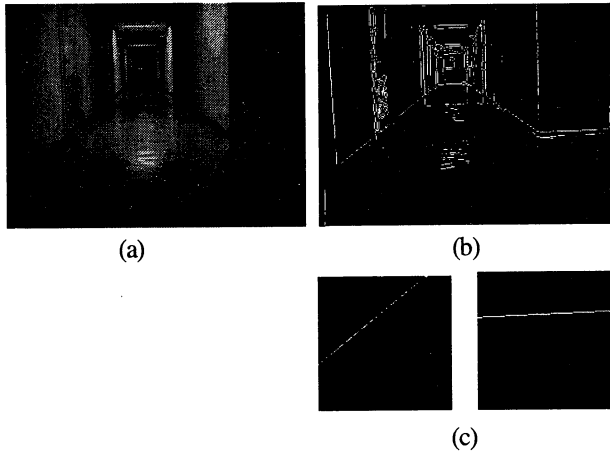
Figure 5: images when the wheelchair faces a turning point on right-hand side. (a) a picture image. (b) a binary image after tinning. (c) left and light images after Hough transform.

(3) Facing a Wall

If a wheelchair faces a wall, two image regions capture a same boundary line or only one image region captures a boundary line as shown in Figure 6. Therefore,

$a_L \approx a_R$ or $a_L > 0$ or $a_R < 0$. (9)

Then the system pans the camera to find a corridor and turns the wheelchair to face a corridor.
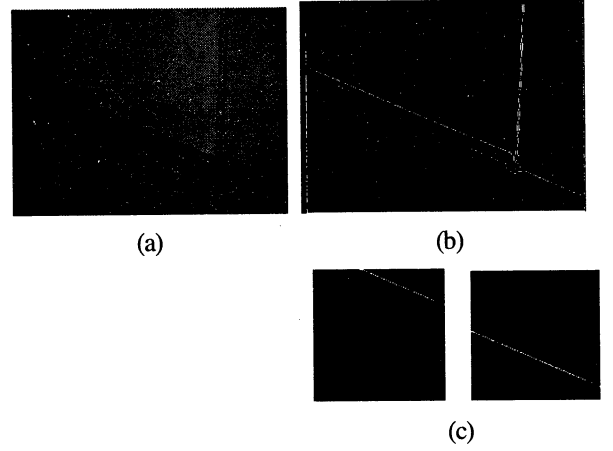
Figure 6: images when the wheelchair faces a wall in the front right direction. (a) a picture image. (b) a binary image after tinning. (c) left and light images after Hough transform.

5 Experiments

5.1 Experimental Environment

We investigate wheelchair navigation on the second floor in the north building of the faculty of engineering, Miyazaki University as shown in Figure 7. The wheelchair starts at the point S, runs through the points A, B, C, D and A and returns to the point S.

In this experiment we investigate autonomous runs of the wheelchair in the going strait region (A), in the right turning region (B), in the 180° turning region (C) and in the T-junction region (D).
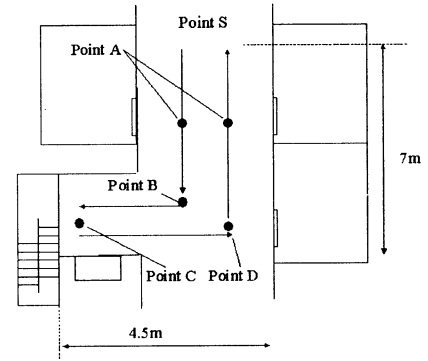
Figure 7: Experimental environments.

5.2 Results

(1) Going straight region

Figure 8(a) and 8(b) show trajectories of the wheelchair which runs from the start point S to the turning point B and runs from the T-junction region D to the start point S. The wheelchair has a tendency to turn left when going straight. However it recognizes a wall and turns right due to avoid colliding with a wall, because the wheelchair system controls the wheelchair so

that it captures both boundary lines in left and right image regions.

(a)   (b)
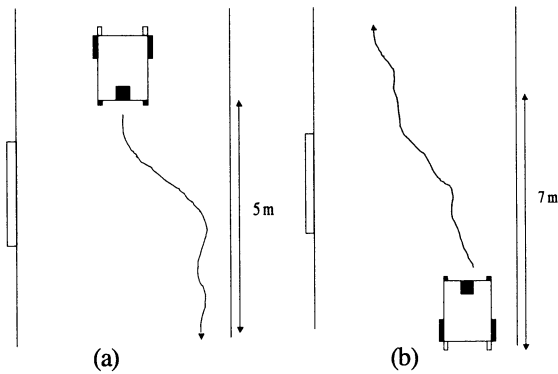
Figure 8: Trajectories of the wheelchair. (a) From the start point S to the turning point B. (b) From the T-junction region D to the start point S.

(2) Right turning region

The wheelchair system recognizes the turning point B about 2 m before. Then the system pans the camera to the turning direction. If image captured by the camera has both boundary lines of a corridor, the system pans the camera to the front direction and turns the wheelchair to the turning direction.
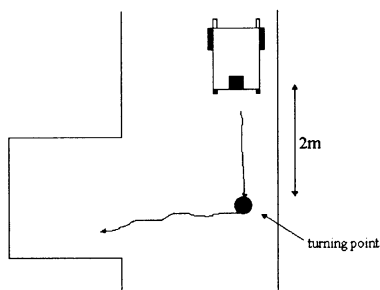
Figure 9: Trajectory of the wheelchair around the point B.

(3) 180° turning region

When the wheelchair faces a wall, the system pans the camera to the left and right directions. Then the system recognizes a corridor in the right direction at the point ① and ②, so that it turns the wheelchair to the right direction.
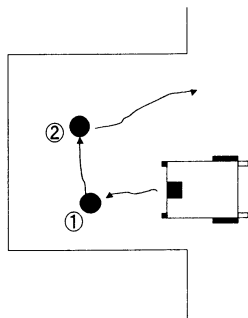
Figure 10: Trajectory of the wheelchair around the point C.

(4) T-junction region

When the wheelchair faces a wall, the system pans the camera to the left and right directions. Then the system recognizes a corridor in the left direction.
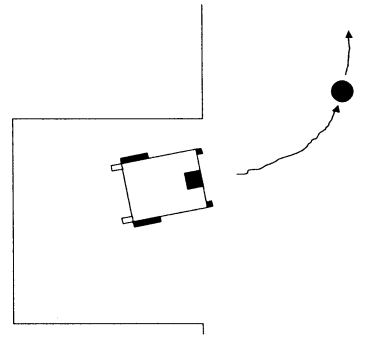
Figure 11: Trajectory of the wheelchair around the point D.

6 Conclusion

We have proposed navigation system for an electric wheelchair using a CCD camera. This system has used a single camera and 2 dimensional picture images to perceive the surrounding environments because 3 dimensional image processing has large amount of calculation. In our experiment, we have found that navigation system recognizes a boundary line of a corridor and a wall properly and that the wheelchair navigates without colliding with a wall.

As the future works, we will develop the wheelchair so that we can ride on it and run safely and comfortably.

References

[1] D.P. Miller, M.G. Slack, "Design and Testing of a Low-Cost Robotic Wheelchair Prototype", *Autonomous Robotics*, 2, pp.77-88, 1995.

[2] T. Gomi, A.Griffith, "Developing Intelligent Wheelchairs for the Handicapped", *Assistive Technology and Artificial Intelligence, Lecture Notes in AI*, 1458, pp.150-178, Springer, 1998.

[3] Y. Matsumoto, T. Ino, T. Ogasawara, "Development of Intelligent Wheelchair System with Face and Gaze Based Interface", *Int. Workshop on Robot and Human Interactive Communication*, pp.262-267, 2001.

[4] S. Nakanishi, Y. Kuno, N. Shimada, Y. Shirai, "Robotic Wheelchair Based on Observations of Both User and Environment", *Int. Conf. on Intelligent Robotics and Systems*, pp912-917, 1999.

Wheelchair Navigation Systems with Infrared Sensors

Tetsuro Kitazoe*, Masayoshi Tabuse*, Tsuyoshi Uemura*, Satoshi Kitazato*, Tatsuro Shinchi**

*Department of Computer Science and Systems Engineering, Faculty of Engineering

**The Center of Educational Research and Practices, Faculty of Education and Culture

Miyazaki University, 1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192, Japan

{kitazoe@cs, tabuse@cs, cs9909@cs, cs9809, shin@edc}.miyazaki-u.ac.jp

Abstract

In this paper, we propose wheelchair navigation system with infrared sensors to aid senior people and the handicapped to drive a wheelchair. We consider two types of control methods of this system. One is If-Then-Else rule. The other is neural network model. We investigate that a wheelchair controlled by this system runs safely and comfortably by neural network model.

1 Introduction

Recently, many people have been researching autonomous mobile robots extensively. One purpose of this research is to build the robots that are able to run safely in unknown or dynamically changing environments. In order to perceive the surrounding environments, robots mount many kinds of sensors, e.g., infrared sensors, ultrasonic range sensors and CCD cameras.

On the other hand, as the number of senior people has been increasing, there is a growing demand for more safe and comfortable wheelchairs as mobile aids. There are two research issues with these wheelchairs. One is on autonomous (or semi-autonomous) and safe navigation, such as avoiding obstacles, wall following, going to a goal using various sensors [1,2]. The other is on human interfaces for easy operation [3,4].

In this paper, we propose wheelchair navigation system with infrared sensors to aid senior people and the handicapped to drive a wheelchair. In this system, a PC gets infrared sensor values and controls an electric wheelchair. We consider two types of control methods of this system. One is If-Then-Else rule. The other is neural network model. We investigate that a wheelchair controlled by this system runs safely and comfortably.

2 Outline of Wheelchair system

The wheelchair system in our research is composed of a commercial electric wheelchair (Matsunaga MD-100), a desktop PC (CPU:PentiumII 400MHz, OS:Linux) and infrared sensors (GP2D12) as shown in Figure 1. The wheelchair has the maximum speed of 4.5 [km/h]. The infrared sensors are mounted on the front and right-hand side. The PC processes infrared sensor values and controls motors of the wheelchair.

The infrared sensor outputs a distance to an object in voltage. Figure 2 shows distance – voltage graph for several kinds of objects. We find that the maximum detection range of the sensor is about 80 cm and has almost same responses to them.
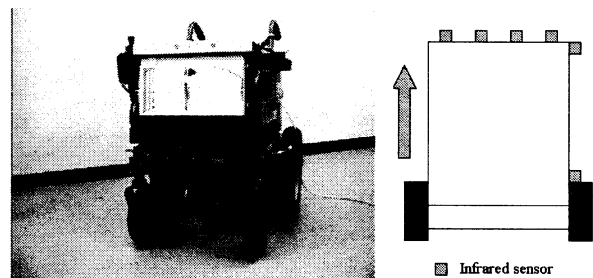
Figure 1: Wheelchair system
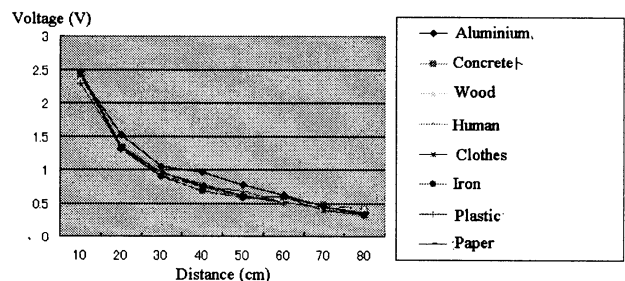
Figure 2: Distance – Voltage graph of a infrared sensor.

3 Control methods

We consider control methods of a wheelchair so that a wheelchair runs along a right side wall. We mount infrared sensors on the wheelchair as shown in Figure 3. Two sensors (x4, x5) are installed on front and rear right-hand side, because the wheelchair runs along a right side wall. Four sensors (x0, x1,

x2, x3) detect an obstacle and a wall in the front direction, so that a front sensor value is represented by the average of these four sensor values X_0

$$X_0 = \{\frac{x_0 + x_1 + x_2 + x_3}{4}\} \quad (1)$$

and other sensor values are denoted by

$$X_1 = x_4, \quad (2)$$

$$X_2 = x_5. \quad (3)$$
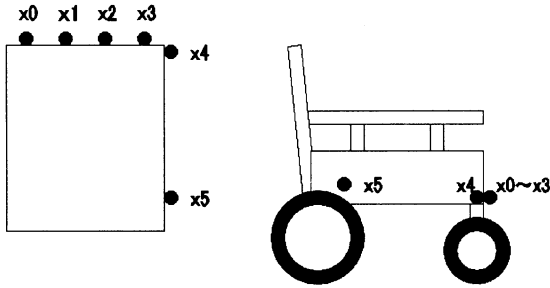
Figure 3: Positions of six infrared sensors.

3.1 If-Then-Else Rule

First we consider If-Then-Else rules for the wheelchair to run along a right-hand side wall. We set a threshold such that if a sensor value is greater (smaller) than the threshold, we treat each sensor value $X_i, i = 0,1,2$ as 1 (0) for simplicity. Thus all sensor states are 8 and each state is shown in Fig. 4. We also choose three kinds of wheel speeds as $V_{R,L} = V_0, 0, -V_0$ and express them as $R, L = +1, 0, -1$, respectively. We determine If-Then-Else rule for these states as follows.

(1) $(X_0, X_1, X_2) = (0,0,0)$, $(L,R) = (1,0)$

Any sensor does not detect a wall, so that the wheelchair turns right widely to approach the wall.

(2) $(X_0, X_1, X_2) = (0,0,1)$, $(L,R) = (1,0)$ *

Only the rear sensor detects a wall and the wheelchair runs in the front left direction. So the wheelchair turns right widely to run in parallel with the wall.

(3) $(X_0, X_1, X_2) = (0,1,0)$, $(L,R) = (0,1)$

Only the front side sensor detects a wall and the wheelchair runs in the front right direction. So the wheelchair turns left to avoid a collision with the wall and to run in parallel with the wall.

(4) $(X_0, X_1, X_2) = (0,1,1)$, $(L,R) = (1,1)$

Two side sensors detect a wall and wheelchair runs in parallel with the wall. So the wheelchair runs straight.

(5) $(X_0, X_1, X_2) = (1,0,0)$, $(L,R) = (-1,1)$

Front sensors detect a wall in front, so that the wheelchair makes a quick left turn to avoid a collision with the wall.

(6) $(X_0, X_1, X_2) = (1,0,1)$, $(L,R) = (-1,1)$

Front sensors and the rear sensor detect a wall, so that the wheelchair turns left quickly to avoid a collision with the wall.

(7) $(X_0, X_1, X_2) = (1,1,0)$, $(L,R) = (-1,1)$

Front sensors and the front side sensors detect a wall, so that the wheelchair makes a quick left turn to avoid a collision with the wall.

(8) $(X_0, X_1, X_2) = (1,1,1)$, $(L,R) = (-1,1)$

All sensors detect a wall, so that the wheelchair makes a quick left turn to avoid a collision with the wall.

The wheelchair controls given by Fig.4 are easily expressed by If-Then-Else rule and coded in PC. Therefore, PC gets sensor values through AD board and gives control signals to left and right wheels through DA board.
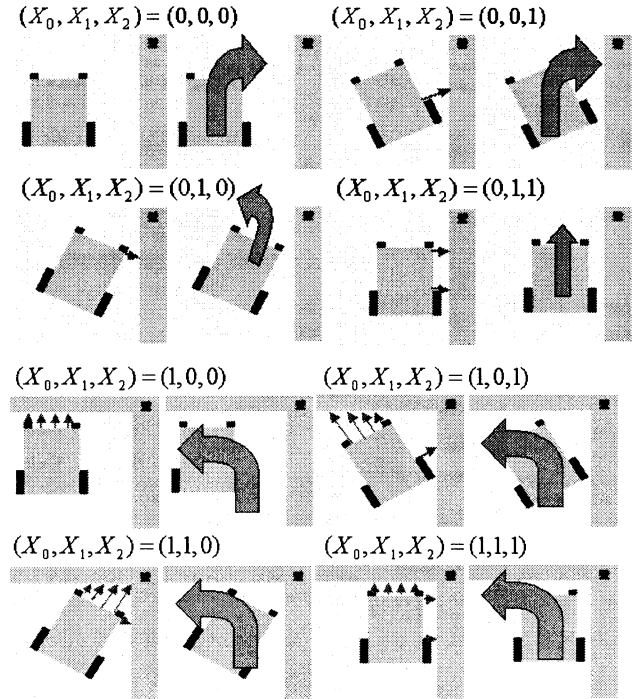
Figure 4: 8 sensor states and the wheelchair behaviors.

3.2 Neural Network Model

The If-Then-Else model given in the previous subsection can be easily transformed to neural network model. The wheel controls

given in the previous subsection are expressed in a three dimensional space (X_0, X_1, X_2) as shown in Fig.5, where speeds of each wheel are expressed by block circle for maximum speed, triangle for zero speed and white circle for maximum backward rotation. If one introduces a step function defined by

$$f(x) = 1 \quad \text{for} \quad x > 0$$
$$ 0 \quad \text{for} \quad x = 0 \qquad (4)$$
$$ -1 \quad \text{for} \quad x < 0,$$

Then If-Then-Else rule model is expressed as

$$L = f(-4X_0 - X_1 + X_2 + 1)$$
$$R = f(X_1 + X_2) \qquad (5)$$

The reason is explained as that the flat surface $-4X_0 - X_1 + X_2 + 1 = 0$ in eq.(5) separates positive and negative regions in Fig.5 and correctly gives wheel speed for left wheel. Similarly right wheel speed is expressed by the second equation of eqs.(5).

We are now at the stage to extend If-then-Else model to neural network model. If we introduce smoothly increasing function

$$F(x) = \tanh(wx) \qquad (6)$$

and extend the former equations (5) to

$$L = F(-4X_0 - X_1 + X_2 + 1)$$
$$R = F(X_1 + X_2) \qquad (7)$$

We can control wheel speeds in a smooth way. Equation (7) tells us that the wheel speed becomes larger if the point (X_0, X_1, X_2) is far away from the flat surfaces where the arguments of F in eq.(4) is zero. In this sense we can control wheel speeds in an analog way. Furthermore we can take original real sensor values for X_i, instead of taking digital values 0 or 1 for X_i, so that a set of sensor values distributes inside of left and right cubes given by Fig.5. Thus, we conclude that we have arrived at purely analogy model. The equation (7) is called as a neural network model whose general forms are expressed by

$$R = f(\sum_{i=0}^{5} W_{Ri} x_i + W_{R6})$$

$$L = f(\sum_{i=0}^{5} W_{Li} x_i + W_{L6}) \qquad (8)$$

and shown in Fig.6. The learning algorithm is expressed as follows. When a flat space is moved a little bit from the surface

shown in Fig.5 by changing synaptic W_i, we learn that the new synaptic couplings are better than the former if the wheelchair has a better performance than the former and vice versa. Therefore, we can obtain a learning algorithm in which a better performance is obtained step by step by measuring an appropriate evaluation function.

L=1 for ●   R=1 for ●
L=0 for △   R=0 for △
L=-1 for ○
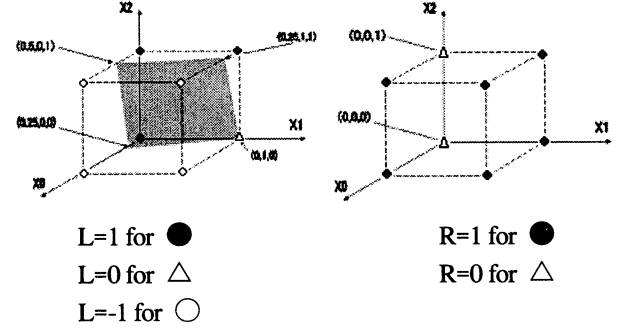
Figure 5: Three dimensional space.
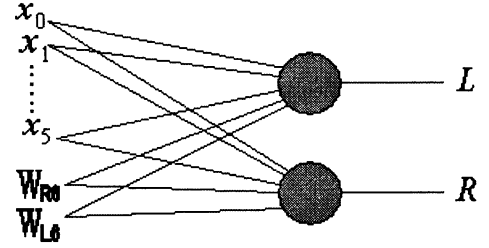
Figure 6: Neural network model.

4 Experiments
4.1 Experimental Environment

We make three experiments on wheelchair navigation on the second floor in the north building of the faculty of engineering, Miyazaki University as shown in Figure 7. Three experiments are as follows.

Ex.1: Experiment with a straight region.
Ex.2: Experiment with a turning point.
Ex.3: Experiment with a concave wall.
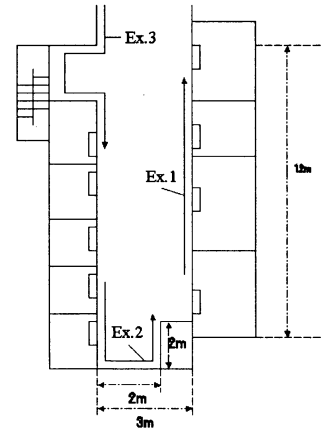
Figure 7: Experimental environment

4.2 Results

Fig. 8(a), 8(b), 8(c) show Ex.1, Ex.2, Ex.3 for the If-Then-Else rule model, respectively and Fig. 9(a), 9(b), 9(c) show Ex.1, Ex.2, Ex.3 for the neural network model, respectively. We understand from the Figures that the If-Then-Else rule model gives behaviors very much depending on the initial conditions and circumstances around the wheelchair. Different pictures in the same Figure in Fig.(8) shows the results for different initial conditions. The neural network model, on the other hand, keeps its trail in a stable and smooth manner and has robustness under noisy environment.

5 Conclusions

A wheelchair navigation system is considered by using infrared sensors. Two types of control models are considered. One is If-Then-Else model and the other is neural network model. It was discussed that neural network model has superior behavior than the If-Then-Else model under the complicated circumstances.

(a)

(b)
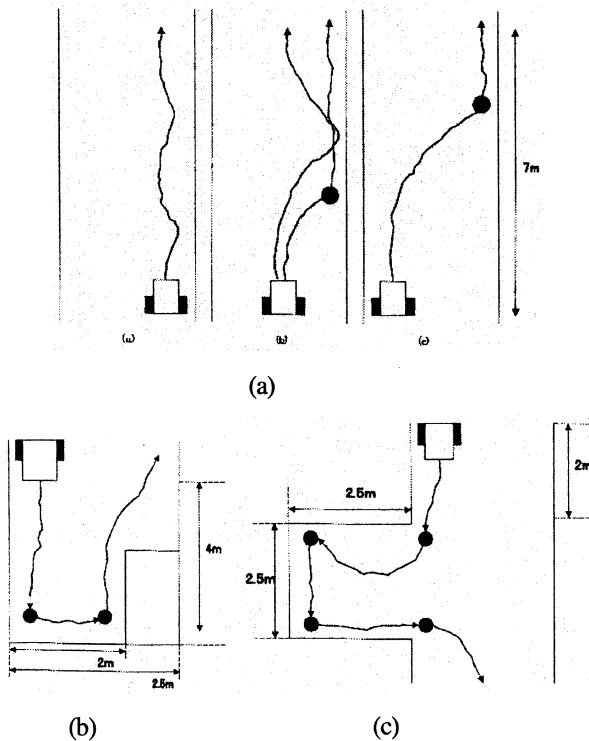
Figure 8: Trails of the wheelchair controlled by If-Then Else rules. (a) a straight region. (b) a turning point (c) a concave wall.

(a)

(b)

(c)
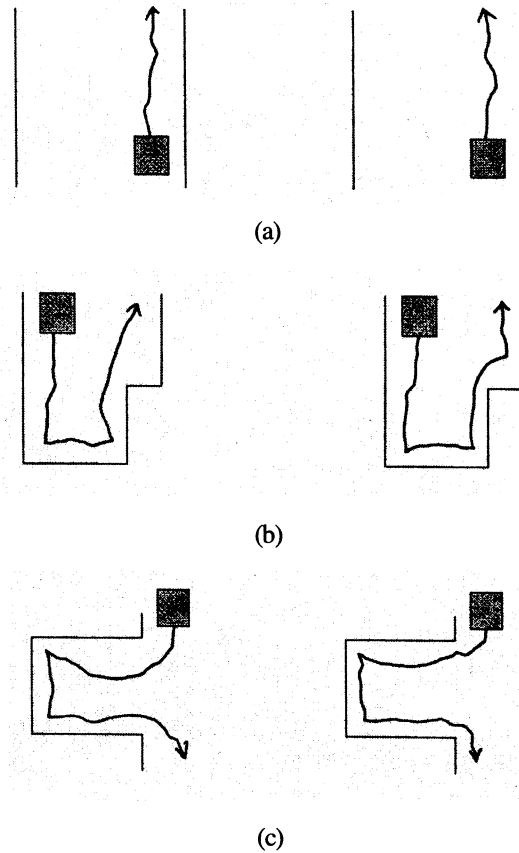
Figure 9:Trails of the wheelchair controlled by If-Then-Else rules and neural network model are shown in left pictures and in right pictures, respectively. (a) a straight region. (b) a turning point. (c) a concave wall.

References

[1] D.P. Miller, M.G. Slack, "Design and Testing of a Low-Cost Robotic Wheelchair Prototype", *Autonomous Robotics*, 2, pp.77-88, 1995.

[2] T. Gomi, A.Griffith, "Developing Intelligent Wheelchairs for the Handicapped", *Assistive Technology and Artificial Intelligence, Lecture Notes in AI*, 1458, pp.150-178, Springer, 1998.

[3] Y. Matsumoto, T. Ino, T. Ogasawara, "Development of Intelligent Wheelchair System with Face and Gaze Based Interface", *Int. Workshop on Robot and Human Interactive Communication*, pp.262-267, 2001.

[4] S. Nakanishi, Y. Kuno, N. Shimada, Y. Shirai, "Robotic Wheelchair Based on Observations of Both User and Environment", *Int. Conf. on Intelligent Robotics and Systems*, pp912-917, 1999.

Similarity-based Image Retrieval System Using PIFS Codes

Takanori Yokoyama   Toshinori Watanabe   Ken Sugawara

Graduate School of Information Systems
University of Electro-Communications
Chofugaoka 1-5-1, Chofu-shi, Tokyo, 182-8585 JAPAN

Abstract

We propose a new retrieval system of images using PIFS coding. In PIFS encoding, a compression code contains mapping information between similar regions in the same image. These mapping information can be treated as vectors, and representative vectors can be generated using them. Representative vectors describe the feature of the image. Hence, the similarity between images is calculable from representative vectors directly. This similarity is applicable to image retrieval. In this report, we explain this scheme and demonstrate its possibility experimentally.

1 Introduction

A lot of image retrieval system has already been studied, and some of them are used commercially[1]. These many systems use the database of original image set. However, there are a few systems which used the database of compressed image set directly.

In this paper, we suggest a new retrieval technique that uses compression codes, especially we use fractal image compression method here. This compression is a relatively recent technique and based on the self-similarity of images. Focusing on a robust property of this compression method, we have developed new similarity retrieval system.

2 Related Work

In general, image retrieval process can be described as follows. At first, a system does feature extraction from an original image set and creates indices of their image sets. These indices are registered with a database and treated as objects.

In retrieval, the system extracts the feature of a query image which a user requested, and compares the feature of the query image with the database. As a result, the system shows several candidate image to the user. In this process, a performance of the system greatly depends on an process of feature extraction from original image sets.

If a system uses the database of compressed codes, A system do not extract feature from original image set . Compression operation is reducing redundancy from original data[2]. And we can consider this process corresponds to feature extraction process from original image sets.

Based on the same concept, the retrieval technique that used Wavelet and DCT exists[1]. However, there is no report about a retrieval system which use fractal image compression method.

3 Similarity-based Image Retrieval

3.1 Fractal Image Compression

Fractal Image Compression devised by Michel F. Barnsley. The main idea of this coding an image is the observation that self-similarity is found within images and is extractable. This employs an affine transforms. An iterated function system consists of a set of these transforms. Therefore, Fractal Image Compression called as an IFS[3].

Barnsley's method was no automated encoding algorithm. Jacquin improved Barnsler's method and developed a automatic compression method[4].

As shown in Figure.1, Jacquin defined large regions called *domain* that are associated with smaller regions called *range*. Each range region is defined as being similar with a domain region. This method is called PIFS coding because of partitioning a image to regions[5].

This compression technique can decode the image from the code arbitrary resolution. From this property, The system can create a image of arbitrary resolution according to the situtation.

Figure 1: Similar Regions. Small regions are *range* and large regions are *domain*.

3.2 Representative Vectors

In the PIFS code, Relations of similarity regions are recorded as followed,

$$\begin{bmatrix} rx \\ ry \\ rz \end{bmatrix} = w_i \begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} \qquad (1)$$

where, rx and ry represent position of the range region, and rz representsa brightness value of this region. dx, dy and dz represent position information and a brightness value of domain region, too. These domain and range regions are similarity relation. This relational is represented by a contractive mapping w_i.

As mentioned before, the PIFS code itself can be considered as a feature of the image. In addition, we introduce the following process as Figure 2 for effective specification.

1. Extraction of mapping vectors from the code
   The system extracts corresponding range region(R) and domain region(D) and obtain a vector from (R) to (D).

2. Clustering for mapping vectors
   In process (1), a lot of vectors are generated. By clustering these vectors, we obtain a new vector set. This is a representative mapping vector set and we can consider it as a specific feature of this image.

In our proposed method, we treat the representative vector as the feature vector of the image and apply to the image retrieval system.

In next subsection, this process is described in detail.

3.3 Similarity Using Correlation of PIFS Codes

As mentioned before, fractal image compression searches for domain regions which are similar to the range regions. And we can characterize the image by mapping vectors from the range to the domain. Important points of this method are

1. A relation between the domain and the range does not change so long as the image does not change greatly.

2. Mapping vectors of a image is different from those of the other image in case that these images are completely different.

Similarity between two images is calculated by following process. Here we denote two images as image A and image B, and denote vector sets in each image as A and B.

1. Decision of a vector pair
   A vector is selected from set A, and corresponding vector that satisfies the following formula is searched in set B. This process is executed for all vectors in A.

$$b_i = f(a_i) = \arg\min_{b \in B} \|a - b\| \qquad (2)$$

2. Calculation of the similarity
   From a correspondence of vector sets, The similarity of two images are calculated by following formula.

$$s(A, B) = \frac{|f(A)|}{|A|} \qquad (3)$$

where $|f(A)|$ is the number of corresponding vector of $f(a) \in B$. And $|A|$ is cardinal number of set A.

From this definition, the range of the similarity is between 0 and 1. It means that if the two images are completely different, the value of s becomes 0, and if they are same image, this value becomes 1.

3.4 Distance between two images

In retrieval, we used the d_{sim} distance included the similarity. The distance $d(A, B)$ to be provided when the system decides a correspondence vector is represented as follows.

$$d(A, B) = \sum_{i=1}^{k} \min_{b \in B} \|a_i - b\| \qquad (4)$$

(a) Original Image   (b) Mapping Vectors   (c) Representative Mapping Vectors
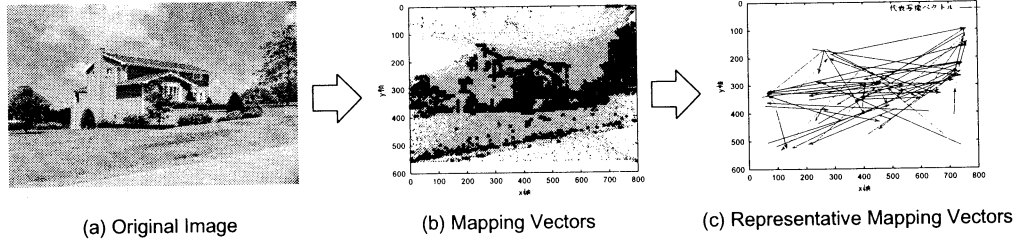
Figure 2: Generating a set of representative mapping vectors

If two images are similar, this accumulation distance becomes a small value. Using this distance and the similarity, we suggest the following distance.

$$d_{sim}(A, B) = d(A, B)\frac{1}{s(A, B)} \qquad (5)$$

This distance prevents a false retrieval when a similarity is high by mistake.

4 Experiments

To confirm the effectiveness of our proposed method, we carried out two types of fundamental experiments. The experiments were executed by a PC with dual CPU(Pentium3-550MHz) and its OS was Linux 2.2.17.

4.1 Robustness of the proposed method

At first, we confirmed the robustness of this system by changing size, angle, brightness and noise. Figure 3 is the relation between the magnification and the similarity. This figure shows the similarity is not so different if the size is changed. Especially, the value keeps high until the magnification is 1.6 times. We also confirmed the robustness for angle, brightness and noise.

4.2 Image retrieval

Next, we will show the result for image retrieval. We prepared a database composed of 960 digital images(8 bit gray scale, 320x240 pixel), which contains natural sceneries, buildings, dolls, and so on.

They are preprocessed and described by representative vectors. Table 1 shows the result for retrieval. Images in the left column are the query images, and 4 images in right column are the searched images obtained by the proposed method.

(a) Representative mapping vectors every magnification rate   (b) Similarities every magnification rate
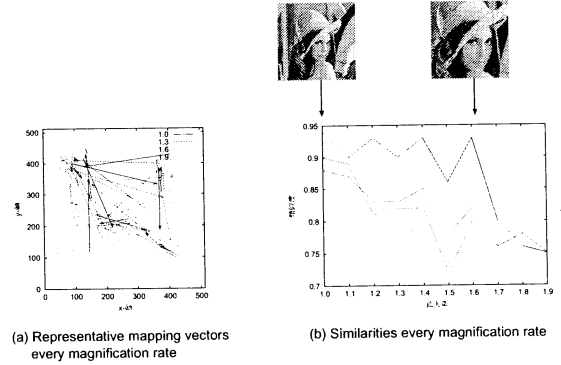
Figure 3: Similarity for magnifications of the image
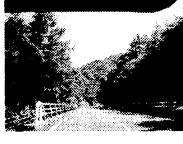
Values in each image are the distance between the query images and each of them. This result shows the proposed method works well as an image retrieval system.

5 Conclusion

We proposed new method for image retrieval using PIFS. In our method, we get the mapping relations between the range region and domain region in an image, and we described the relations by vectors from the range to the domain at first. Next, some representative vectors are generated by using them. They are treated as the feature vectors of the image. In retrieval process, we calculated the similarity between the images by defining the similarity and the distance between two images. The effectiveness of the proposed method was confirmed by a fundamental experiments.

Now we are evaluating the efficiency of the system using large scale database.

Table 1: Retrieval Results

Query Image	Rank 2	Rank 3	Rank 4	Rank 5
0.0	49.1	55.8	56.1	56.4
0.0	50.6	51.7	53.5	56.0
0.0	57.6	58.0	64.6	65.6
0.0	48.4	69.1	70.8	74.7

References

[1] T. Huang, Y. Rui(1997), Image Retrieval: Past, present, and future, International symposium on Multimedia Information Processing.

[2] David Salomon(2000), Data compression: the complete reference, Springer-Verlag New York.

[3] Martin J. Turner, Jonathan M. Blackledge(1998), Fractal Geometry in Digital Imaging. Academic Press.

[4] Brendt Wohlberg, Gerhard de Jager(1999), A Review of the Fractal Image Coding Literature, IEEE Transactions on Image Processing, vol. 8, no. 12, December 1999.

[5] Yuval Fisher(1995), Fractal image compression: theory and application, Springer-Verlag New York.

[6] Takanori Yokoyama, Toshinori Watanabe, Ken Sugawara and et al(2002), A Structural Similarity Extraction Method using PIFS code(In Japanese), Technical Report of IEICE, PRMU2001-285, pp.105-112, 2002.

[7] Takanori Yokoyama, Toshinori Watanabe, Ken Sugawara(2002) Feature Extraction and Retrieval of Images Using Correlation of PIFS Codes(In Japanese), ITE Techonical Report, Vol. 26, No. 54, pp.29-32, 2002.

Stabilization of LTI Systems with Periodic Communication Constraints by Output Sampled Hold Control

N. Takahashi
Faculty of Enginnering
Miyazaki University
1-1 Gakuenkibanadai Nishi,
Miyazaki, Japan, 889-2192.
takahasi@cs.miyazaki-u.ac.jp

M. Kono
Faculty of Enginnering
Miyazaki University
1-1 Gakuenkibanadai Nishi
Miyazaki, Japan, 889-2192.
kouno@cs.miyazaki-u.ac.jp

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 output sampled hold control. This method has the merit that capacity of an sensor-controller communication bus is little.

keywords: linear control system, periodic systems, limited communication control, sampled hold control

1 Introduction

For the problem of remote control system in which sensors and actuators are connected via network, it is important how to communicate signals under constrained capacity of communication bus. Such a constraints cause bad influence to system performance. In [2, 3], [9], 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 or observer based controller.

Several authors represented by Araki et al. have dealt with stabilization of time-invariant continuous systems by time-varying output feedback [4]. On the other hand, Kaczorek proposed the similar method for discrete-time systems [5]. We extend this method to the periodic discrete-time case independently of Kaczorek, and call it output sampled hold control [6, 8]. In [8], under the assumption that eigenvalues of a monodromy matrix are distinct, we have given a necessary and sufficient condition that the close loop system can be stabilized.

In this paper, we apply the result of [8] to stabilization of LTI system with periodic communication constraints. Our method has a merit that capacity of a sensor-controller communication bus is little.

2 Preliminaries

Let the plant be an LTI (linear time-invariant) discrete time system

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t), \end{aligned} \quad (1)$$

where $x \in \mathcal{R}^n$ is a state vector, $u \in \mathcal{R}^m$ is a actuator-input vector, $y \in \mathcal{R}^p$ 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 elements of the controller output are transmitted to the plant. Only the corresponding elements of the actuator-input are updated and other elements are held until next signals arrive. Similary, only partial elements of the sensor-output are transmitted to the controller.

We assume that the controller-plant communication follows 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 are 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 similary. $\sigma_i(t)$ and $\sigma_o(t), (t = 0, 1, 2, \ldots)$ are called communication sequences.

Definition 1 *Assume that capacity of the input (output) bus is constrained by $b_c < m (b_o < p)$. A pair $(\sigma_c(t), \sigma_o(t))$ is admissible if the following conditions are satisfied,*

i) $\|\sigma_c(t)\|^2 \le b_c, \|\sigma_o(t)\|^2 \le b_o, t = 0, \ldots, \omega - 1$

*ii) $span\{\sigma_c(0), \sigma_c(1), \ldots, \sigma_c(\omega - 1)\} = \mathcal{R}^m$
$span\{\sigma_o(0), \sigma_o(1), \ldots, \sigma_o(\omega - 1)\} = \mathcal{R}^p$*

where $\| \cdot \|$ denotes the Euclidean norm.

3 Derivation of periodic systems

We assume that the pair $(\sigma_c(t), \sigma_o(t))$ is admissible. Let $y_\iota(t)$ be a controller input signal at time t, then

$$y_\iota(t) = diag(\sigma_o(t))y(t)$$

$$+ (I - \mathrm{diag}(\sigma_o(t)))y_\iota(t-1), \qquad (2)$$

where $\mathrm{diag}(x)$ denotes a diagonal matrix, whose diagonal element is x_i. By substituting iteratively, we have

$$\begin{aligned}
y_\iota(t) = \ & \mathrm{diag}(\sigma_o(t))y(t) + (I - \mathrm{diag}(\sigma_o(t))) \\
& \cdot \mathrm{diag}(\sigma_o(t-1)y(t-1) + (I - \mathrm{diag}(\sigma_o(t))) \\
& \cdot((I - \mathrm{diag}(\sigma_o(t-1)))\mathrm{diag}(\sigma_o(t-2))y(t-2) \\
& + \cdots + (I - \mathrm{diag}(\sigma_o(t)))(I - \mathrm{diag}(\sigma_o(t-1))) \\
& \cdots \mathrm{diag}(\sigma_o(t-\omega+1))y(t-\omega+1), \qquad (3)
\end{aligned}$$

Now we define

$$\begin{aligned}
D_o(t,t) = \ & \mathrm{diag}(\sigma_o(t)) \\
D_o(t,s) = \ & (I - \mathrm{diag}(\sigma_o(t)))(I - \mathrm{diag}(\sigma_o(t-1))) \cdots \\
& \cdot (I - \mathrm{diag}(\sigma_o(t-s+1)))\mathrm{diag}(\sigma_o(s)), \\
& s = t-1, \ldots, t-\omega+1. \qquad (4)
\end{aligned}$$

Then equation (3) can be expressed as

$$y_\iota(t) = \sum_{s=t-\omega+1}^{t} D_o(t,s)y(s). \qquad (5)$$

Let $u_\iota(t)$ be a controller-output signal. Then, by the similar procedure, we get

$$u(t) = \sum_{s=t-\omega+1}^{t} D_c(t,s)u_\iota(s), \qquad (6)$$

where

$$\begin{aligned}
D_c(t,t) = \ & \mathrm{diag}(\sigma_o(t)) \\
D_c(t,s) = \ & (I - \mathrm{diag}(\sigma_c(t)))(I - \mathrm{diag}(\sigma_c(t-1))) \cdots \\
& \cdot (I - \mathrm{diag}(\sigma_c(t \quad s+1)))\mathrm{diag}(\sigma_c(s)), \\
& s = t-1, \ldots, t-\omega+1 \qquad (7)
\end{aligned}$$

By substitutng (6) to (1), we get

$$x(t+1) = Ax(t) + B\sum_{s=t-\omega+1}^{t} D_c(t,s)u_\iota(s). \qquad (8)$$

The second term in the right hand of equation (8) becomes

$$\begin{aligned}
& \sum_{s=t-\omega+1}^{t} BD_c(t,s)u_\iota(s) \\
& = BD_c(t,t)u_\iota(t) + [BD_c(t,t-1), \ldots, \\
& \qquad BD_c(t,t-\omega+1)]
\begin{bmatrix} u_\iota(t-1) \\ u_\iota(t-2) \\ \vdots \\ u_\iota(t-\omega+1) \end{bmatrix}.
\end{aligned}$$

It follows from (1) and (5) that

$$\begin{aligned}
y_\iota(t) = \ & \sum_{s=t-\omega+1}^{t} D_o(t,s)Cx(s) \\
= \ & [D_o(t,t)C, D_o(t,t-1)C, \ldots, \\
& D_o(t,t-\omega+1)C]
\begin{bmatrix} x(t) \\ \vdots \\ x(t-\omega+1) \end{bmatrix}. \quad (9)
\end{aligned}$$

Define

$$x_a(t) := \begin{bmatrix} x(t) \\ x(t-1) \\ \vdots \\ x(t-\omega+1) \\ u_\iota(t-1) \\ u_\iota(t-2) \\ \vdots \\ u_\iota(t-\omega+1) \end{bmatrix},$$

$$n_a := \omega(m+n) - m.$$

Then, we have

$$x_a(t+1) = A_a(t)x_a(t) + B_a(t)u_\iota(t), \qquad (10)$$

$$A_a(t) = \begin{bmatrix} A_{a_{11}}(t) & A_{a_{12}}(t) \\ 0 & A_{a_{22}}(t) \end{bmatrix} \qquad (11)$$

$$A_{a_{11}}(t) = \begin{bmatrix} A & 0 & \ldots & 0 \\ I & & & \\ & & & \\ 0 & \ldots & I & 0 \end{bmatrix},$$

$$A_{a_{12}}(t) = \begin{bmatrix} BD_c(t,t-1) \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$$A_{a_{22}}(t) = \begin{bmatrix} 0 & & & 0 \\ I_m & & & \\ & \ddots & & \\ 0 & & I_m & 0 \end{bmatrix}.$$

$$B_a(t) = \begin{bmatrix} BD_c(t,t) \\ 0 \\ \vdots \\ 0 \\ I_m \\ 0 \\ \vdots \\ 0 \end{bmatrix} u_\iota(t). \qquad (12)$$

Further, let

$$\begin{aligned}
C_a(t) = \ & [D_o(t,t)C, \ldots, \\
& D_o(t,t-\omega+1)C | 0, \ldots, 0]. \qquad (13)
\end{aligned}$$

Then it follows from (9) that

$$y_\iota(t) = C_a(t)x_a(t). \qquad (14)$$

Let $\sum_e$ denote the periodic extended system (10) and (14). Let

$$\begin{aligned}
X_a(t_0) = \ & X_a(t_0, t_0 - \omega) \\
= \ & A_a(t_0-1) \cdots A_a(t_0-\omega) \\
= \ & \begin{bmatrix} X_{a_{11}}(t_0) & X_{a_{12}}(t_0) \\ 0 & X_{a_{22}}(t_0) \end{bmatrix} \qquad (15)
\end{aligned}$$

Then, it follows from (11) that

$$X_{a_{11}}(t_0) = A_{a_{11}}^{\omega} \tag{16}$$

$$X_{a_{22}}(t_0) = A_{a_{22}}^{\omega} = 0 \tag{17}$$

$$\begin{aligned}
X_{a_{12}}(t_0) &= A_{a_{11}}^{\omega-1} A_{a_{12}}(t_0 - \omega) \\
&\quad + A_{a_{11}}^{\omega-2} A_{a_{12}}(t_0 - \omega + 1) A_{a_{22}} \\
&\quad + A_{a_{11}}^{\omega-3} A_{a_{12}}(t_0 - \omega + 2) A_{a_{22}}^2 \\
&\quad \cdots + A_{a_{11}} A_{a_{12}}(t_0 - 2) A_{a_{22}}^{\omega-2}
\end{aligned} \tag{18}$$

Remark 1 $X_a(t_0)$ *is called a monodromy matrix of* $\sum_e$ *at time* t_0 *and its eigenvalues are independent of* t_0.

The following proposition shows a the special property of the monodromy matrix for $\sum_e$.

Proposition 1 $X_a(t_0)$ *has* $(m+n)(\omega-1)$ *zero eigenvalues.*

The proof of **Proposition 1** is given in Appendix. For system $\sum_e$, we set the following definitions.

Definition 2 *Let an eigenvalue and a right eigenvector of* $X_a(t)$ *be* λ *and* $w(t)$ *respectively. The eigenvalue* λ *is said to be sample-unobservable at time* t, *if* $C(t)w(t) = 0$.

Remark 2 *System* $\sum_e$ *is said to be sample-observable at time* t, *if all eigenvalues of* $X_a(t)$ *are sample-observable.*

Definition 3 *System* $\sum_e$ *is said to be sample-detectable at time* t, *if all the eigenvalues such that* $|\lambda| \geq 1$ *are sample-detectable.*

In order to apply the result of [6], we introduce the time-invariant system $\sum_a$ assosiated with $\sum_e$.

$$\sum_a :$$
$$\begin{aligned}
&x_a((l+1))\omega + t_0) \\
&= X_a(t_0)x_a(l\omega + t_0) + Q_a(t_0)[u^T(l\omega + t_0), \\
&\qquad \ldots, u^T(l\omega + \omega - 1 + t_0)]^T,
\end{aligned} \tag{19}$$

$$y(l\omega + t_0) = C_a(t_0)x(l\omega + t_0), \tag{20}$$

where

$$\begin{aligned}
Q_a(t_0) &= [X_a(t_0, t_0 - \omega + 1)B_a(t - \omega), \\
&\qquad X_a(t_0, t_0 - \omega + 2)B_a(t_0 - \omega + 1), \\
&\qquad \ldots, B_a(t_0 - 1)].
\end{aligned} \tag{21}$$

The following proposition gives the relation between $\sum_a$ and $\sum_e$.

Proposition 2 $\sum_e$ *is sample-detectable at time* t_0 *if and only if* $\sum_a$ *is detectable at time* t_0, *and* $\sum_e$ *is reachable at time* t_0 *if and only if* $\sum_a$ *is reachable at time* t_0.

4 Main results

Let a sampled output periodic hold control have the form

$$\begin{aligned}
&u_\iota(t) = H(t)y_\iota(in_a\omega + t_0), \\
&H(t + n_a\omega) = H(t), \\
&t \in [in_a\omega + t_0, (i+1)n_a\omega + t_0), \\
&\qquad i = 0, 1, \ldots.
\end{aligned} \tag{22}$$

By applying control (22) to system $\sum_e$, the state transition of the closed-loop system from time $in_a\omega + t_0$ to time $(i+1)n_a\omega + t_0$ obeys

$$\begin{aligned}
x_a((i+1)n_a\omega + t_0) &= \Psi_{t_0}x(in_a\omega + t_0), \\
&\qquad i = 0, 1, \ldots
\end{aligned} \tag{23}$$

where

$$\begin{aligned}
\Psi_{t_0} &= X_a(n_a\omega + t_0, t_0) + \sum_{j=t_0+1}^{n_a\omega+t_0} X_a(n_a\omega + t_0, j) \\
&\qquad \times B(j-1)H(j-1)C_a(t_0).
\end{aligned} \tag{24}$$

Stability of the closed-loop system is dominated by the eigenvalues of Ψ_{t_0}. That is, the system (23) is asymptotically stable if and only if all eigenvalues of Ψ_{t_0} have magnitudes less than 1. The following theorem is a main result of this paper.

Theorem 1 *There exist a time-varying matrix* $H(t)$ *such that the closed-loop system (23) is asymptotically stable only if*

  i) $\sum_e$ *is stabilizable at time* t_0

  ii) $\sum_e$ *is sample-detectable at time* t_0

Proof Let $\lambda, v^T(t_0)$ and $w(t_0)$ denote an eigenvalue, a left eigenvector and a right eigenvector of $X_a(\omega + t_0, t_0) = X_a(t_0)$, respectively.

Proof of (i) : Suppose that system (1) is not stabilizable at t_0. Then, there exist λ and $v^T(t_0) \neq 0$ such that $|\lambda| \geq 1$ and

$$v^T(t_0)X_a(\omega + t_0, t_0) = \lambda v^T(t_0) \tag{25}$$

$$\begin{aligned}
&v^T(t_0)X_a(\omega + t_0, j)B(j-1) = 0, \\
&\qquad j = t_0 + 1, t_0 + 2, \ldots, t_0 + \omega.
\end{aligned} \tag{26}$$

It follows from (25) and (26) that

$$\begin{aligned}
&\sum_{j=t_0+1}^{n_a\omega+t_0} v^T(t_0)X_a(n_a\omega + t_0, j)B(j-1)H(j-1)C_a(t_0) \\
&= \sum_{q=0}^{n_a-1}\sum_{k=1}^{\omega} v^T(t_0)X_a(n_a\omega + t_0, q\omega + k + t_0) \\
&\qquad \times B(q\omega + k - 1 + t_0)H(q\omega + k - 1 + t_0)C_a(t_0) \\
&= \sum_{q=0}^{n_a-1}\sum_{k=1}^{\omega} v^T(t_0)X_a(n_a\omega + t_0, (q+1)\omega + t_0) \\
&\qquad \times X_a((q+1)\omega + t_0, q\omega + k + t_0)
\end{aligned}$$

$$
\begin{aligned}
&\times B(q\omega + k - 1 + t_0)H(q\omega + k - 1 + t_0)C_a(t_0) \\
&= \sum_{q=0}^{n_a-1}\sum_{k=1}^{\omega} v^T(t_0)X_a((n_a - q - 1)\omega + t_0, t_0) \\
&\quad \times X_a(\omega + t_0, k + t_0) \cdot B_a(k - 1 + t_0) \\
&\quad \times H(q\omega + k - 1 + t_0) \cdot C_a(t_0) \\
&= \sum_{q=0}^{n_a-1}\sum_{k=1}^{\omega} v^T(t_0)(X_a(\omega + t_0, t_0))^{(n_a-q-1)} \\
&\quad \times X_a(\omega + t_0, k + t_0) \cdot B_a(k - 1 + t_0) \\
&\quad \times H(q\omega + k - 1 + t_0) \cdot C_a(t_0) \\
&= \sum_{q=0}^{n_a-1}\sum_{k=1}^{\omega} \lambda^{(n_a-q-1)}v^T(t_0)X_a(\omega + t_0, k + t_0) \\
&\quad \times B_a(k - 1 + t_0) \cdot H(q\omega + k - 1 + t_0) \cdot C_a(t_0) \\
&= 0. \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad (27)
\end{aligned}
$$

Since premultiplying (24) by $v^T(t_0)$ gives

$$
\begin{aligned}
v^T(t_0)\Psi_{t_0} &= v^T(t_0)X_a(n_a\omega + t_0, t_0) + \sum_{j=t_0+1}^{n_a\omega+t_0} v^T(t_0) \\
&\quad \times X_a(n_a\omega + t_0, j)B_a(j - 1)H(j - 1)C_a(t_0), \quad (28)
\end{aligned}
$$

in virture of (27) we have

$$
\begin{aligned}
v^T(t_0)\Psi_{t_0} &= v^T(t_0)X_a(n_a\omega + t_0, t_0) \\
&= v^T(t_0)X_a(t)^{n_a} = \lambda^{n_a}v^T(t_0). \quad (29)
\end{aligned}
$$

This shows that λ^{n_a} is an eigenvalue of Ψ_{t_0}. Since $|\lambda^{n_a}| \geq 1$, the closed system is not asymptotically stable.

Proof of (ii) : Suppose that $(C_a(t_0), X_a(\omega + t_0, t_0))$ is not sample-detectable. Then, there exist λ and $w(t_0) \neq 0$ such that $|\lambda| \geq 1$ and

$$
\begin{aligned}
X_a(\omega + t_0, t_0)w(t_0) &= \lambda w(t_0), \quad\quad (30) \\
C_a(t_0)w(t_0) &= 0. \quad\quad\quad\quad (31)
\end{aligned}
$$

Postmultiplying (24) by $w(t_0)$ gives

$$
\begin{aligned}
\Psi_{t_0}w(t_0) &= X_a^{n_a}(t_0)w(t_0) \\
&= \lambda^{n_a}w(t_0), \quad |\lambda^{n_a}| \geq 1 \quad (32)
\end{aligned}
$$

This shows that the closed-loop system is not asymptotically stable.

Q.E.D.

5   Conclusion

We have given a necessay condition for LTI systems with periodic communication constraints to be stabilizable by output sample hold control. The problem for future study is to show that the condition of **Theorem 1** is also sufficient. Then, it is expected that $\sum_a$ plays an important role.

References

[1] W. Zhang, M. S. Branicky and S. M. Pillips (2001). Stability of Network Control Systems. IEEE Control System Magazine. 21(1):84–99

[2] D. Hristu (2000). Stabilization of LTI Systems with Communication Constraints. Proc. 2000 American Control Conference. 2342–2346

[3] D. Hristu and K. Morgansen (1999). Limited commucination control. Systems & Control Letters. 37:193–205

[4] T. Hagiwara and M. Araki (1988). Time-Varying Digital Controllers. Journal of the society of instrument and control engineers(In Japanese). 27(12):1071–1077

[5] T. Kaczorek (1985). Pole Placement for Linear Discrete time Systems by Periodic Output Feedback. Systems & Control Letters. 6(4):pp.267–269

[6] T. Suzuki and M. Kono (1994). Stabilization of Discrete time periodic system by output sampled hold control. Trans. of the Society of Instrument and Control Engineering (In Japanese). 30(8):pp.987–989

[7] T. Suzuki and M. Kono (1998). Transition Matrix Assignment for Periodic Disctrete System via Output Feedback (In Japanese). Trans. of the Society of Instrument and Control Engineering. 34(9):1153–1158

[8] M. Kono, T. Suzuki, N. Takahashi and O. Sato (1999). Stabilization of Periodic Discrete Systems by Output Sample Hold Control (In Japanese). Trans. Institute of Systems, Control and Information Engineers. 12(4):pp.220–224

[9] S. Yamamoto, T. Kawagoe and T. Ushio (2002). Stabilization of Control Systems with a Periodic Communication Constraint by an Observer-based Controller (In Japanese). Trans. Institute of Systems, Control and Information Engineers. 15(11):pp.593-599

A   Appendix

A.1   Proof of Proposition 1

Since

$$
A_{a_{11}}^{\omega} = \begin{bmatrix} A^{\omega}, & 0, & ,\ldots \\ A^{\omega-1}, & 0, & ,\ldots \\ \vdots & \vdots & \\ A, & 0, & ,\ldots \end{bmatrix},
$$

in virture of (16), we have

$$
\text{rank}X_{a_{11}}(t_0) = n. \quad\quad (33)
$$

It is clear that $X_a(t_0)$ has $(m + n)(\omega - 1)$ zero eigenvalues.

Q.E.D.

Computational Complexity for the Simulation of Four-Dimensional Marker Automata by Four-Dimensional Turing Machines

Hidenobu Okabe
Dept. of Comp. Sci. and
Sys. Eng., Miyazaki Univ.,
Miyazaki, 889-2192 Japan

Makoto Sakamoto
Dept. of Comp. Sci. and
Sys. Eng., Miyazaki Univ.,
Miyazaki, 889-2192 Japan

Katsushi Inoue
Dept. of Comp. Sci. and
Sys. Eng., Yamaguchi Univ.,
Ube, 755-8611 Japan

Abstract

We think that recently, due to the advances in computer animation, virtual reality systems and so forth, it is useful for analyzing computational complexity of multi-dimensional information processing to explicate the properties of four-dimensional automata. In this paper, we investigate the computational complexity for the simulation of four-dimensional one-marker automata by seven-way four-dimensional Turing machines.

1 Introduction

Blum, et al. proposed two-dimensional automata as the computational model of two-dimensional pattern processing, and investigated their pattern recognition abilities[1]. After that, due to the advances in many application areas such as computer vision, robotics, and so on, the study of three-dimensional automata has been meaningful[3]. However, in recent years, there have arisen many requirements for dynamic image precessing with advances in computer animation, virtual reality systems, and so forth. Thus, we think that it is useful for analyzing dynamic images to study the properties of four-dimensional automata, i.e., three-dimensional automata with the time axis as the computational model of dynamic image processing. This paper deals with four-dimensional one-marker automata in terms of the space complexities that seven-way four-dimensional Turing machines, which can move east, west, south, north, up, down, and future, but not past on four-dimensional rectangular input tapes, require and suffice to simulate four-dimensional one-marker automata.

2 Preliminaries

Definition 2.1. Let Σ be a finite set of symbols. A *four-dimensional tape* over Σ is a four-dimensional rectangular array of elements of Σ. The set of all four-dimensional tapes over Σ is denoted by $\Sigma^{(4)}$. Given a tape $x \in \Sigma^{(4)}$, for each $j(1 \leq j \leq 4)$, we let $l_j(x)$ be the length of x along the jth axis. When $1 \leq i_j \leq l_j(x)$ for each $j(1 \leq j \leq 4)$, let $x(i_1, i_2, i_3, i_4)$ denote the symbol in x with coordinates (i_1, i_2, i_3, i_4), as shown in Fig. 1. Furthermore, we define

$$x[(i_1, i_2, i_3, i_4), (i'_1, i'_2, i'_3, i'_4)],$$

when $1 \leq i_j \leq i'_j \leq l_j(x)$ for each integer $j(1 \leq j \leq 4)$, as the four-dimensional tape y satisfying the following :

(i) for each $j(1 \leq j \leq 4)$, $l_j(y) = i'_j - i_j + 1$;

(ii) for each r_1, r_2, r_3, r_4 $(1 \leq r_1 \leq l_1(y), 1 \leq r_2 \leq l_2(y), 1 \leq r_3 \leq l_3(y), 1 \leq r_4 \leq l_4(y))$, $y(r_1, r_2, r_3, r_4) = x(r_1 + i_1 - 1, r_2 + i_2 - 1, r_3 + i_3 - 1, r_4 + i_4 - 1)$.
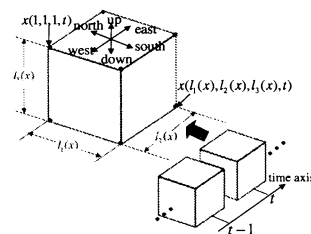
Fig. 1: Four-Dimensional Input Tape.

Definition 2.2. A *four-dimensional nondetermi − nistic one-marker automaton* 4-NM_1 is defined by the six-tuple

$$M = (Q, q_0, F, \Sigma, +, -, \delta),$$

where

(1) Q is a finite set of *states*,

(2) $q_0 \in Q$ is the *initial state*,

(3) $F \subseteq Q$ is the set of *accepting states*,

(4) Σ is a finite input alphabet ($\sharp \notin \Sigma$ is the *boundary symbol*),

(5) $\{+,-\}$ is the pair of signs of presence and absence of the marker,

(6) δ: $((Q \times \{+,-\}) \times ((\Sigma \cup \{\sharp\}) \times \{+,-\})) \rightarrow 2^{(Q \times \{+,-\})} \times ((\Sigma \cup \{\sharp\}) \times \{+,-\}) \times \{$east,west,south,north,up,down,future,past,no move$\})$ is the *next-move function*, satisfying the following: For any $q, q' \in Q$, any $a, a' \in \Sigma$, any $u, u', v, v' \in \{+,-\}$, and any $d \in \{$east,west,south,north,up,down,future,past,no move$\}$, if $((q',u'),(a',v'),d) \in \delta$ $((q,u),(a,v))$ then $a = a'$, and $(u,v,u',v') \in \{(+,-,+,-),(+,-,-,+),(-,+,-,+),(-,+,+,-),(-,-,-,-)\}$.

We call a pair (q,u) in $Q \times \{+,-\}$ an *extended state*, representing the situation that M holds or does not hold the marker in the finite control according to the sign $u = +$ or $u = -$, respectively. A pair (a,v) in $\Sigma \times \{+,-\}$ represents an input tape cell on which the marker exists or does not exsit according to the sign $v = +$ or $v = -$, respectively.

Therefore, the restrictions on δ above imply the following conditions. (A) When holding the marker, M can put it down or keep on holding. (B) When not holding the marker, and (i) if the marker exists on the current cell, M can pick it up or leave it there, or (ii) if the marker does not exist on the current cell, M cannot create a new marker any more.

Definition 2.3. Let Σ be the input alphabet of 4-NM_1 M. An *extended input tape* $\tilde{x}$ of M is any four-dimensional tape over $\Sigma \times \{+,-\}$ such that

(i) for each $j(1 \leq j \leq 4)$, $l_j(\tilde{x}) = l_j(x)$,

(ii) for each $i_1(1 \leq i_1 \leq l_1(\tilde{x}))$, $i_2(1 \leq i_2 \leq l_2(\tilde{x}))$, $i_3(1 \leq i_3 \leq l_3(\tilde{x}))$, and $i_4(1 \leq i_4 \leq l_4(\tilde{x}))$, $\tilde{x}(i_1,i_2,i_3,i_4) = x(i_1,i_2,i_3,i_4,u)$ for some $u \in \{+,-\}$.

Definition 2.4. A configuration of 4-NM_1 $M = (Q,q_0,F,\Sigma,\delta)$ is an element of

$$((\Sigma \cup \{\sharp\}) \times \{+,-\})^{(4)} \times (Q \times \{+,-\}) \times N^4,$$

where N denotes the set of all nonnegative integers. The first component of a configuration $c = (\tilde{x},(q,u),(i_1,i_2,i_3,i_4))$ represents the extended input tape of M. The second component (q,u) of c represents the extended state. The third component (i_1,i_2,i_3,i_4) of c represents the input head position.

If q is the state associated with configuration c, then c is said to be an *accepting configuration* if q is an accepting state. The *initial configuration* of M on input x is

$$I_M(x) = (x^-,(q_0,+),(1,1,1,1)),$$

where x^- is the special extended input tape of M such that $x^-(i_1,i_2,i_3,i_4) = (x(i_1,i_2,i_3,i_4),-)$ for each i_1,i_2,i_3,i_4 $(1 \leq i_1 \leq l_1(\tilde{x}), 1 \leq i_2 \leq l_2(\tilde{x}), 1 \leq i_3 \leq l_3(\tilde{x}), 1 \leq i_4 \leq l_4(\tilde{x}))$. If M moves determinately, we call M a *four-dimensional deterministic one-marker automaton* 4-DM_1.

Definition 2.5. A *seven-way four-dimensional Turing machine* is defined by the six-tuple

$$M = (Q,q_0,F,\Sigma,\Gamma,\delta),$$

where

(1) Q is a finite set of *states*,

(2) $q_0 \in Q$ is the *initial state*,

(3) $F \subseteq Q$ is the set of *accepting states*,

(4) Σ is a finite *input alphabet* ($\sharp \notin \Sigma$ is the *boundary symbol*),

(5) Γ is a finite *storage-tape alphabet* ($B \in \Gamma$ is the *blank symbol*), and

(6) $\delta \subseteq (Q \times (\Sigma \cup \{\sharp\}) \times \Gamma) \times (Q \times (\Gamma - \{B\}) \times \{$east,west,south,north,up,down,future,no move$\} \times \{$right,left,no move$\})$.

If M moves determinately (nondeterminately), we call M a *seven-way four-dimensional deterministic (nondeterministic) Turing machine* SV4-DTM (SV4-NTM).

Let L: $N^3 \rightarrow R$ be a function. A seven-way four-dimensional Turing machine M is said to be $L(l,m,n)$ *space bounded* if for each $l,m,n \geq 1$ and for each x with $l_1(x) = l$, $l_2(x) = m$ and $l_3(x) = n$, if x is accepted by M, then there is an *accepting computation path* of M on x in which M uses no more than $L(l,m,n)$ cells of the storage tape. We denote an $L(l,m,n)$ space bounded SV4-DTM (SV4-NTM) by SV4-DTM($L(l,m,n)$) (SV4-NTM($L(l,m,n)$)).

Definition 2.6. Let $T(M)$ be the set of four-dimensional tapes accepted by a machine M, and let $\pounds[4$-$DM_1] = \{T | T(M)$ for some 4-DM_1 $M\}$. $\pounds[4$-$NM_1]$, etc. are defined in the same way as $\pounds[4$-$DM_1]$.

We can easily derive the following theorem by using ordinary technique[3].

Theorem 2.1. *For any function* $L(l,m,n) \geq \log(l mn)$, $\pounds[SV4$-$NTM(L(l,m,n))] \subseteq U_{c>0}$ $\pounds[SV4$-DTM $(2^{c(L(l,m,n))})]$.

3 Sufficient Spaces

In this section, we investigate the sufficient spaces (i.e., upper bounds) for seven-way four-dimensional Turing machines to simulate four-dimensional one-marker automata.

Theorem 3.1. $\mathcal{L}[4\text{-}DM_1]$
$\subseteq \mathcal{L}[\text{SV4-NTM}(lmn\log(lmn))].$

Proof : Suppose that a 4-DM_1 $M = (Q, q_0, F, \Sigma, \delta)$ is given. We partition the extended states $Q \times \{+, -\}$ into disjoint subsets $Q^+ = Q \times \{+\}$ and $Q^- = Q \times \{-\}$ which correspond to the extended states when M is holding and not holding the marker in the finite control, respectively. We assume that M has a unique accepting state q_a, i.e., $|F| = 1$. In order to make our proof clear, we also assume that M begins to move with its input head on the southmost and eastmost bottom boundary symbols $\sharp$'s of input tape at time t, i.e., position $(l + 1, m + 1, n + 1, t + 1)$ and, when M accepts an input, it enters the accepting state at the same position $(l+1, m+1, n+1, t+1)$ with the marker held in the finite control.

Suppose that an input tape x with $l_1(x) = l$, $l_2(x) = m$, $l_3(x) = n$ and $l_4(x) = t$ is given to M. For M and x, define three types of functions $f_h^{\uparrow -}$, $f_h^{\uparrow +}$ and $f_h^{\downarrow -}$.

$f_h^{\uparrow -}(q^-, i, j, k) = (q'^-, i', j', k')$: Suppose that we make M start from the configuration $(x^-, q^-, (i,j,k,h-1))$, i.e., no marker existing either on the input x or in the finite control of M. After that, if M reaches the hth three-dimensional rectangular array of x in some time, the configuration corresponding to the first arrival is $(x^-, q'^-, (i',j',k',h))$,

$f_h^{\uparrow +}(q^+, i, j, k) = (q'^+, i', j', k')$: Suppose that we make M start from the configuration $(x^-, q^+, (i,j,k,h-1))$, i.e., holding the marker in the finite control of M. After that, if M reaches the hth three-dimensional rectangular array of x with its marker held in the finite control in some time (so, when M puts down the marker on the way, it must return to this position again and pick up the marker), the configuration corresponding to the first arrival is $(x^-, q'^+, (i',j',k',h))$,

$f_h^{\downarrow -}(q^-, i, j, k) = (q'^-, i', j', k')$: Suppose that we make M start from the configuration $(x^-, q^-, (i,j,k,h+1))$, i.e., no marker existing either on the input tape or in the finite control of M. After that, if M reaches the hth three-dimensional rectangular array of x in some time, the configuration corresponding to the first arrival is $(x^-, q'^-, (i',j',k',h))$,

l: M never reaches the hth three-dimensional rectangular array of x.

Then, we can show that there exists an SV4-

NTM($lmn\log(lmn)$) M' such that $T(M')=T(M)$. Roughly speaking, while scanning from the top three-dimensional rectangular array down to the bottom three-dimensional rectangular array of the input along the time axis, M' guesses $f_h^{\downarrow -}$, constructs $f_{h+1}^{\uparrow -}$ and $f_{h+1}^{\uparrow +}$, checks $f_{h-1}^{\downarrow -}$, and finally at the bottom three-dimensional rectangular array of the input, M' decides by using $f_{t+1}^{\uparrow -}$ and $f_{t+1}^{\uparrow +}$ whether or not M accepts x. In order to record these mappings for each h, $O(lmn)$ blocks of $O(\log(lmn))$ size suffice, so in total, $O(lmn\log(lmn))$ cells of the working tape suffice. It will be obvious that $T(M)=T(M')$. $\square$

From Theorem 2.1 and 3.1, we get the following.

Corollary 3.1. $\mathcal{L}[4\text{-}DM_1]$
$\subseteq \mathcal{L}[SV4\text{-}DTM(2^{O(lmn\log(lmn))})].$

We next show that $l^2m^2n^2$ space is sufficient for SV4-NTM's to simulate 4-NM_1's. The basic idea of the proof are the same as those of Theorem 3.1.

Theorem 3.2. $\mathcal{L}[4\text{-}NM_1] \subseteq \mathcal{L}[SV4\text{-}NTM(l^2m^2n^2)].$

From Theorem 2.1 and 3.1, we get the following.

Corollary 3.2. $\mathcal{L}[4\text{-}NM_1]$
$\subseteq \mathcal{L}[SV4\text{-}DTM(2^{O(l^2m^2n^2)})].$

4 Necessary Spaces

In this section, we show that the algorithms described in the previous section are optimal in some sense.

Definition 4.1. Let x be in $\Sigma^{(4)}$ (Σ is a finite set of symbols) and $l_1(x) = l$, $l_2(x) = m$ and $l_3(x) = n$. For each r ($1 \leq r \leq Q[l_4(x)/lmn]$) (where $Q[l_4(x)/lmn]$ denotes the quotient when $l_4(x)$ is divided by lmn),

$$x[(1, 1, 1, (r-1)lmn + 1), (l, m, n, rlmn)]$$

is called the rth (l, m, n)-block of x. We say that the tape x has exactly $c(l, m, n)$-blocks if $l_4(x) = clmn$, where c is a positive integer.

Definition 4.2. Let $(l_1, m_1, n_1), (l_2, m_2, n_2), \ldots$ be a sequence of points (i.e., pairs of three natural numbers), and let $\{(l_i, m_i, n_i)\}$ denote this sequence. We call a sequence $\{(l_1, m_1, n_1)\}$ the *regular sequence of points* if $(l_i, m_i, n_i) \neq (l_j, m_j, n_j)$ for $i \neq j$.

Lemma 4.1. *Let* $T_1 = \{x \in \{0,1\}^4 | \ ^{\exists}l \geq 1, \ ^{\exists}m \geq 1,$ $^{\exists}n \geq 1 [\ l_1(x)=l \ \& \ l_2(x)=m \ \& \ l_3(x)=n \ \& \ (each three-dimensional rectangular array of x contains exactly*

one "1") & $^\exists d{\geq}2[(x$ *has exactly* d (l,m,n)-*blocks,* *i.e.,* $l_4(x) = dlmn$) & (*the last* (l,m,n)-*blocks is equal* *to some other* (l,m,n)-*block*)]]}. *Then,*

(1) $T_1 \in \mathcal{L}[4\text{-}DM_1]$, *but*

(2) $T_1 \notin \mathcal{L}[SV4\text{-}DTM(2^{L(l,m,n)})]$ *(so,* $T_1 \notin \mathcal{L}[FV4\text{-}$ $NTM(L(l,m,n))])$ *for any function* $L(l,m,n)$ *such that*

$$lim_{i\to\infty}[L(l_i,m_i,n_i)/(l_im_in_i\log(l_im_in_i))] = 0$$
for some regular sequence of points $\{(l_i,m_i,n_i)\}$.

Proof : (1) We construct a 4-DM_1 M accepting T_1 as follows. Given an input x with $l_1(x)=l$, $l_2(x)=m$, $l_3(x)=n$ and $l_4(x)=t$, M first checks, by sweeping three-dimensional rectangular array by three-dimensional rectangular array, that each three-dimensional rectangular array of x contains exactly one "1," and M then checks, by making a zigzag of 45°-direction from top three-dimensional rectangular array to bottom three-dimensional rectangular array, that x has exactly d (l,m,n)-blocks for some integer $d \geq 2$. After that, M tests by utilizing its own marker whether the last (l,m,n)-block is identical to some other (l,m,n)-block: In order to check whether the pth three-dimensional rectangular array of the hth (l,m,n)-block is identical to the pth three-dimensional rectangular array of the last (l,m,n)-block ($1{\leq}p{\leq}lmn$, $1{\leq}h{\leq}d$), M first puts the marker on the position $(i,j,k,lmn(h-1)+p)$. After that, M vertically moves down until encounters the bottom boundary, after which it moves up $(lmn - p)$ three-dimensional rectangular array by making a zigzag of 45°-direction. At this time, M arrives at the pth three-dimensional rectangular array of the last (l,m,n)-block. M then finds the "1" position on the three-dimensional rectangular array and moves up vertically from this position. In this course, each time M meets a "1" position, it checks whether or not there is a marker on the three-dimensional rectangular array (containing the "1" position). In this way, M enters an accepting state just when it finds out some (l,m,n)-block, each of whose three-dimensional rectangular arrays is identical to the corresponding three-dimensional rectangular array of the last (l,m,n)-block. It will be obvious that $T(M)=T_1$.

(2) Suppose to the contrary that there exists an SV4-DTM($2^{L(l,m,n)}$) M accepting T_1, where $L(l,m,n)$ is a function such that

$$lim_{i\to\infty}[L(l_i,m_i,n_i)/(l_im_in_i\log(l_im_in_i))] = 0$$

for some regular sequence of points $\{(l_i,m_i,n_i)\}$. Then, by using the well-known technique[3], we can get the desired result. $\square$

From Lemma 4.1., we can conclude as follows.

Theorem 4.1. *To simulate* $4 - DM_1$'s, (1) *SV4-* *NTM's require* $\Omega(lmn\log(lmn))$ *space and* (2) *SV4-* *DTM's require* $2^{\Omega(lmn\log(lmn))}$ *space.*

Next, we can get the following lemma by using the same technique as in the proof of Lemma 4.1.

Lemma 4.2. *Let* $T_2=\{x \in \{0,1\}^{(4)}|\ ^\exists l \geq 1, ^\exists m \geq 1,$ $^\exists n \geq 1$ [$l_1(x)=l$ & $l_2(x)=m$ & $l_3(x)=n$ & $^\exists d \geq 2$ [(x *has exactly* d (l,m,n)-*blocks, i,e.,* $l_4(x)=dlmn$) & (*the last* (l,m,n)-*block is different from any other* (l,m,n)-*block*)]]}. *Then,*

(1) $T_2 \in \mathcal{L}[4\text{-}NM_1]$, *but* .

(2) $T_2 \notin \mathcal{L}[SV4\text{-}DTM(2^{L(l,m,n)})]$ *(so,* $T_2{\notin}\mathcal{L}[SV4\text{-}$ $NTM(L(l,m,n))])$ *for any function* L *such that* $lim_{i\to\infty}[L(l_i,m_i,n_i)/(l_i^2m_i^2n_i^2] = 0$ *for some* *regular sequence of points* $\{(l_i,m_i,n_i)\}$.

From Lemma 4.2., we can conclude as follows.

Theorem 4.2. *To simulate* 4-NM_1's,

(1) *SV4-NTM's require* $\Omega(l^2m^2n^2)$ *space, and*
(2) *SV4-DTM's require* $2^{\Omega(l^2m^2n^2)}$ *space.*

5 Conclusions

In this paper, we have investigated how much space is necessary and sufficient for seven-way four-dimensional deterministic or nondeterministic Turing machines to simulate four-dimensional deterministic or nondeterministic one-marker automata. It will be interesting to investigate the simulation of four-dimensional "alternating" one-marker atutomata (see [2] for the concept of "alternation").

References

[1] M.Blum, et al., *IEEE Symp.* : 1967.

[2] A.K.Chandra, et al., *J. ACM* 28 (1) : 1981.

[3] M.Sakamoto, et al., *Inform. Sci.* 95 : 1996.

Genetic information processing at Biophysical Models base on Giant DNA chain aggregate by Spermidine, ATP and Mg++ at Biological conditions : Channel Switch Function Formed by Micro & Macro-aggregation

Yasuo Yonezawa[1]*
yonezawa@ipc.ibaraki.ac.jp

Hidetoshi Kuramochi[1,2]
Kuramochi.hidetoshi@nies.go.jp

[1] Graduate school of Science and Engineering, Ibaraki University, 4-12-1 Nakanarusawa, Hitachi, 316-8511, Japan

Abstract

Primary structure generated on Morphologenesis is chromosome as genetic higher structure induced DNA sequences. Metamorphosis of chromosome in cellular nuclear are could be understood as the dynamical topology change depend on several cell growth phases from point of view at structure of packed long DNA chains by several Polyamines species and each concentration ratio of Bio-Molecular . We are well know that the outputs of DNA encode information are request the structural relaxation from compacted double stranded DNA chain in order to the reaction of several biochemical enzymes for DNA replication etc. In this paper deals with the structural channel switch model observed as the phenomena of aggregation and relaxation of long double stranded DNA chains based on experimental results of the DNA aggregate by Biological DNA self-assemble process. From experimental results, we are presented the structural Channel switch model based on DNA aggregation for genetic information processing. Furthermore, we are proposed the development at New Idea of Chromosome like DNA computation based on the structural channel switch model at Long DNA chains.
Keywords: DNA Aggregation, Polyamines, ATP, Mg^{++}, Structural channel Switch, Genomic Information processing.

1 Introduction

DNA is often found a compacted state [1] in the living systems and the manner of packing is closely related to biological functions of DNA, such as replication and transcription [2]. These phenomena are natural DNA computation for maintenance of living activity. Encode information of DNA system are obtained two kind of the sequences code and the time-space decode pattern (decode arrangement) from long DNA chains. One of encode information is well known call as the codon arrangement description (Transcription) in according to "Central dogma." Furthermore, 2nd of encode information is not enough to regulation of decode arrangement from numerous genetic codes. Time-space pattern in order to decode from numerous genetic codes are describes as high structure as Long double stranded chain of compaction states. The previous in intro research has been undertaken in order to deeper insight into procedure of in vivo packing and unfolding. The in vivo compaction is induced by multivalent cation's such

polyamines, hexamine cobalt III (and peptides. At previous several researches for single molecular compactions [3, 4,5 and 6], we are obtained the knowledge against coil globule transition based on single DNA chain (Double stranded DNA molecular). The experiment of multi-molecular environments such as inner-cellular are not enough carried out. In order to elucidate the DNA compaction and un-compaction, we are proposed the Biochemical and Bio-physical chemistry experiment based on Polyamines-DNA self-assemble phenomena. In this investigation, we report that polyamine concentrations induced relaxation (unfolding) of aggregated double stranded DNA chains. Our observed results provided new insight into Metamorphosis phenomena of the structure on long double stranded DNA chains. In other way, linear multivalent polyamines such as spermidine and spermine are abundant in living cells and play a key role in maintained cellular DNA in compact state. Depletion of polyamine revel in vivo inhibits cell growth and interfaces with gene expression. The cellular mechanism of polyamine function is DNA condensation is presumed to involve neutralization of the negatively charges DNA backbone by positively charge amino acid group of spermidine and spermine. In order to elucidate the mechanism constructed with DNA-polyamines aggregation interaction, we are proposed the morphological variety generated with DNA polyamines aggregation process at several conditions. Ever since Adelman's result (7), much speculation has gone into whether or not DNA in fact be used for computation. The running time for a molecular algorithm is proportional to the number of operation on test tubes. The volume is maximum number of strings in all test tubes at any time, counting multiplicities. The strand-length complexity of a molecular algorithm is the length of the longest DNA stranded used in the computation. For the contents of described as above, the preparation time as compared with actual DNA computation time is clear needed the large time. But, Real Living organisms are very fast the actual DNA computation time. Therefore, in order to construction of DNA computation could be quickly preparation of materials for molecular computation, we are investigated soluble for the regulation mechanism of needed information from Long DNA chain in living cells... In this paper, we are presented the Morphological variety such as the channel switch like structure induced with polyamines-DNA self-assemble phenomena

2 Method and Results

2.1 Materials of DNA and Polyamines.

Bacteriophage λDNA (48.5 kbp) dissolved in the TE solution (10mM Tris, 1mM EDTA, pH7.8) was purchased from Takara Shuzou Ltd. and diluted with the

* Corresponding author. TEL:+81-294-38-5208
[2] Present Address: Research Center for Material cycles and Waste Management, National Institute for Environmental Studies, Onogawa 16-2, Tukuba, Ibaraki, 305-0053, Japan

same TE buffer (Sigma, Sigma-Aldrich Co.,) up to appreciate concentration. Before the preparation of sample materials, two tests were carried out as described as described as bellows;

(1) Purity test: Whether any DNA expect for λ DNA detected in the solution was examined by agarose gel electrophoresis [8].

(2) For the inspection protein remain (Contamination), the ratio A260 /A 280 of the absorbance of the DNA solution at 260 nm and 280nm was measured. If single band is observed agarose gel results and the ratio is 1.8 the DNA solution can be used in the further aggregation experiments. The standard spermidine, spermine, Cadaverine and Putrescine are prepared by dissolving each trihydride (Nakarai Tesuque, Inc) into TE buffer.

2.2 Measurements of the amounts of aggregated DNA

This measurement are carried out by the centrifugation assay for DNA aggregation; DNA aggregation was induced the addition of the spermidine, spermine and another polyamines solution to aliquot of the DNA solution, and ultimately the total volume was made to 200 µl. After vortexing for 15sec, the sample were incubated for an hour at room temperature (about 295 k) and centrifuged for 8 min at 11000 ×g by micro-centrifugation (Beckman GS-15R). The supernatant was recovered and analyzed for the determination of DNA concentration in the supernatant by measurement of the absorbance at 260nm. All the absorbance measurements are carried out with Beckman DU700 spectrometer (Used by 350 µg micro quartz cuvette). In these experiments, the amount of DNA remaining in the supernatant was calculated by the absorbance ratio of the supernatant to control solution in the absorbance of each polyamine.

2.3 Observation of the DNA aggregate morphology

Observation by polarizing microscopy: DNA macro-aggregate was recovering from centrifuged micro-tube and deposited between slide and cover slip (these are clean by hydrogen peroxide and rinsed in Milli-Q water and pure ethanol). The cover slip was seals with DPX (Fulka, Sigma-Aldrich Co.,) to prevent dehydration of sample. To analyze the order phase of the aggregate, the sample was observed between crossed poplars under a microscope (Olympus BX60). Macro-scopic imaging by CCD camera: DNA macro-aggregate after centrifugal separation as deposited a plastic transparent sheet (3M GelBond Film), and then was observed by CCD camera (Sony DCR-TRV900).

3. Results

3.1 Spermidine-DNA aggregation process

At several concentrations (3, 24, 48 and 90µg/ml), the amount of aggregate was measured. In the only case of 3 µg/ml DNA, invisible small aggregate was obtained because the light scattering effect was observed due to small particle size. Above 24µg/ml DNA, in contrast, DNA-spermidine complexes really gave rise of a Milli-Meter order segments. The aggregate contents in various conditions were plotted in Figure 1.

The formation of aggregate was not continuous but stepwise, and dependent on the DNA concentrations.
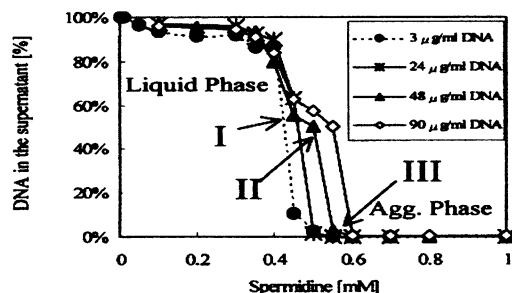
Fig.1 Precipitations curve of the DNA aggregate by spermidine and the morphological variation

The threshold concentration C threshold was maintained at 0.35-0.40 mM spermidine in any case. On the other hand, the concentration for termination of aggregation C term shifted from 0.50-0.60 mM as increasing DNA concentration. In order to examine the self-assembly process, the aggregate morphology observed along the aggregation data. Morphologies of the aggregate at 48µg/ml DNA have been given in Figure 2a-e.

These dramatically varied between 0.45-0.55mM spermidine. At 0.45 mM spermidine, where the aggregate was first recognized as a visible one, Long anisotropic fiber(type I) with some cross over point observed, and much loosely associate compared to the other textures. As consequence of the observation of one hundred fibers (A bundle of Double stranded DNA chain), the average width of the fiber elongated to millimeter order was about 1 µml.
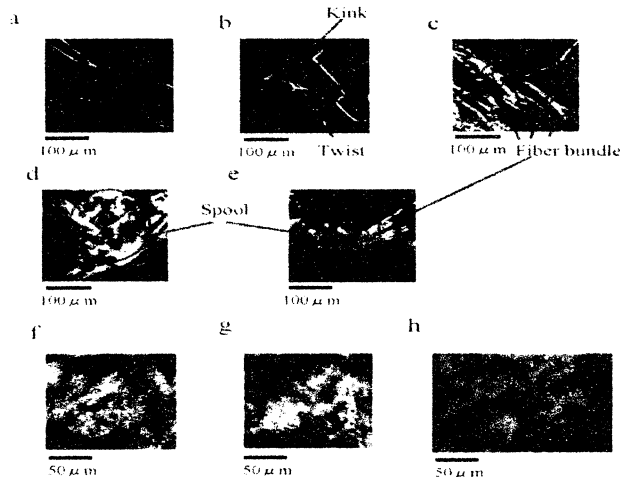
Fig.2 Aspect of DNA macro aggregates in polarizing microscopy

a and b: anisotropic fibers and their kink and twist at the condition in Figure.1

c and d: parallel bundle and spool of anisotropic fibers at the condition in Figure.1

e: cholesteric-like texture at 0.55 mM spermidine at the condition in Figure.1

f:cholesteric-like texture at 1.0 mM spermidine at the condition in Figure.1

g: cholesteric-like texture at 7.0 mM spermidine similar to the observation by Pelta et al.

In this condition, fine fiber 1000 or more (Length/width) is formed. At 0.50 mM, Bondless and spools of these fibers (Type II) were observed (Figure 2a and 2b). The

Bundle and spool implied that the cross-link points increased. The morphological variation from Type II can explain by the quantities change of aggregate DNA. AS shown in Figure 1, the amounts of the aggregate at 0.45mM were roughly equal to that at 0.50 mM. In brief, Increments of spermidine molecule(= 0.50～0.45)was seldom consumed to proceed the formation of renewal DNA fiber. We conclude that between 0.45～0.50mM more spermidine molecules bind to the residual negative site on the fiber and then bound spermidine molecules result in fiber-linkers. After DNA fiber formation, fiber association owing to increase of binding spermidine molecule may be thermo-dynamical preferred to generation further precipitation. At 0.55mM over which the quantity aggregate as the texture no longer changes, the boundless and spool were closely associated that gel-like assembly(Type III) was observed as shown in Figure 2d. The density of this aggregate is much different from the other. It is interesting how influence such drastic difference has enzymatic reaction. From the series of observations, we suggested that the following process accompanying morphological variation as one aggregation pathway (Shown in Figure 3);

200μm

bundles 100μm

200μm

100μm
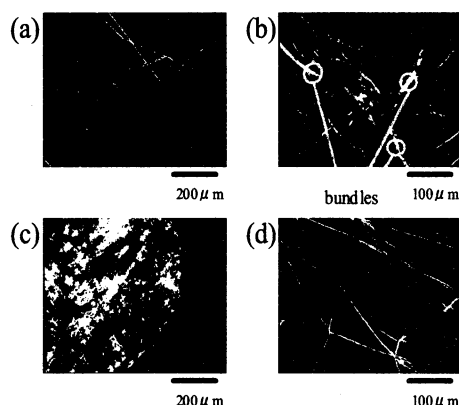
Fig. 3. Polarizing microscopy images of DNA aggregate at various spermidine concentrations. Since DNA aggregate mediated by spermidine has ordered phases[5,7], the aggregate recovered from sample tube was observed between crossed polars in an Olympus BX60 microscope. (a) Fiber shape(Type I.) at 0.45mM spermidine, (b) Bundles of fibers (Type II) at 0.50mM spermidine, (c) A highly condensed phase (Type III) at 0.55mM spermidine, (d) Disaggregation of Type III by thedilution of spermidine concentration to 0.35 mM. In spite of lower than 0.45mM, the aggregate similar to Type 1 is observed. This demonstrates that the aggregating pathway is reversible, but has a hysteresis.

3.2 Effects of Polyamine-DNA aggregation under polyamine's mixture conditions.

In order to elucidate the aggregation phenomena under inner-cellular like biochemical environments, efficiency of DNA aggregation against several polyamines are examined the single and mixture condition of spermidine, spermine, Cadaverine and Putrescine.

Fig. 4-a:DNA micro-aggregate (SEM observation)
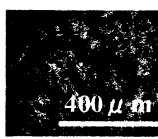
Fig. 4-b:DNA macro-aggregate (PM observation)
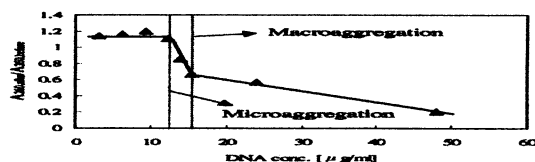
Fig.4-c The ratio of the absorbance's of the DNAsolution

before and after the addition of Spermidine as a function of DNA concentration

The combination ratios of these polyamines are examined with three groups of spermidine+spermine,spermidine+Cadaverine, Spermidine + Putrescine. Each electric charge values of these polyamines are spermidine: +4, spermine: +3, Putrescine and Cadaverine: +2[8]. These experiments under polyamine-mixture condition are indicated the possibility of regulation of DNA aggregation depend on the ratio of each polyamines concentration in mixture solution described as bellows:
(1) Observation of aggregation at Spermine (+4) and Spermidine (+3) mixture.: Aggregation phenomena increased with Spermine and spermidine than spermine only(shown in Figure 5).
(2)Observation of aggregation with Putresine(+2).Cadaverine(+2) and Spermidine(+3): +2 Polyamines are inhibited spermidine aggregation(shown in Figure 5)
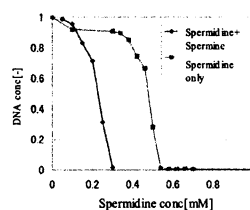
Fig.5-a:Agregation effect of Spermidine + Spermine. In case of Spermidine 0.35 mM addition.
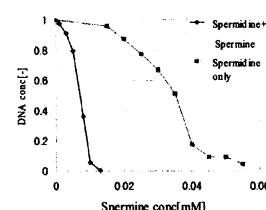
Fig.5-b:Aggregation effect of Spermine + Spermidine. In case of Spermine 0.015 mM addition.
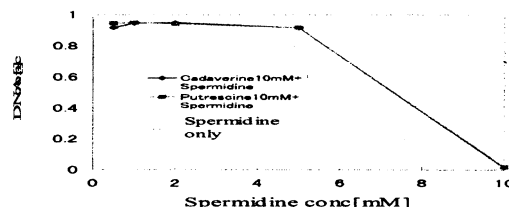
Fig.5-c:Aggregation effect of polyamine(+2)

3 Discussions and Perspective for Experiment Results

In previous research at interaction of DNA polyamines, we are well known that the each cell proliferation phase at several organisms' species is dependent on the supply of polyamines to divisional cells. Decreasing cellular polyamines significantly inhibit cell division, but the involved in growth inhibition remains to elucidate. Although exact role for cellular polyamines in specific biochemical events related to cell proliferation at the molecular revel are largely unknown, several research have indicated that expression of gene is at least partially involved in the early modulation of cell growth stimulation by polyamines. Introduction of cell growth in vivo as well as in vitro is accompanied by a significant increase in gene expression after an increase in cellular polyamines, which precedes the indication of DNA synthesis. All the observation concluded that long double stranded DNA molecules undergo the following three-stage process in the macro-aggregation:
(1) Anisotropy fiber with cross-over point, (2) Parallel bundles and spools, (3) Assemble with their multi-layer
    The morphological variation, Self-assembly process,

and analogy of aggregate to chromosome structure under cell proliferation states in spermidine induced DNA macro-aggregation have been first experimental results. Furthermore, the depletion state (empty state) of DNA-spermidine aggregate is very similar with the D.D.Dunlap et al report [4]. These behavioral phenomena of DNA aggregation were observed as morphological variety in to these experimental results, it was particularity well suited for attracting condensates from three-dimensional solute phase, and trapping them in two dimensions with the least possible distortion of their native structures. The observed morphological variety are suggested the possibility of channel switch for decoding of the information of necessary protein et al at optimal timing for cell growth by relaxation of aggregate long DNA chains. In this research results, we are harvests the essential process of polyamines-DNA aggregation described as bellows contents:

(1) Compaction of DNA chain: Packing and encode information
(2) Relaxation of DNA chain: decode Information
(3) Regulation of Compaction and Relaxation speed and region

These natural polyamines (Putrescine, Cadaverine, Spermidine, and Spermine) are well known to stabilize pyrimidine-purine/prymidine and purine-purine-pyrimidine triplex DNA formation. In general, penta-amines were more efficacious than tetra-amines in stabilizing triplex DNA chain, although most the polyamines with pendant free amino group caused DNA aggregation below 50% conversion to triplex DNA.

From previous research results, we could be obtained the next contents described as bellows; these results provide new insight into mechanism of DNA precipitate by polyamines. Long Double stranded DNA chain condensed into handled, folded loop of DNA likely reflect that most condensation proceeds through folding rather than winding DNA. The concentrations of polyamines are increased markedly upon stimulation of RNA synthesis. At present, the biological significance of resolution observed at higher polyamines concentration is not enough clear, because the concentration required (generally about 50mM) appear un-physiological (the polyamines intra-cellular concentrations are mM range: general average about 10mM). It is possible, however, that exit locally high concentrations of polyamine that are involved in un-condensing (relaxation) rather condensing process or that the concentration required for an in vivo effects is lower than in vitro conditions. In these facts, we focused that the highest concentrations in polyamines are generally detected cell growth phase (for example: G1 phase of cell growth cycle) [9]. Therefore, the suggestion has been made that these high concentrations are required in the inner-cells preparation for DNA synthesis. Futhermore, this research can also be discussed at speculation of biological generative complexity [10], we are considerate at model of channel switch from the relaxation and compaction by aggregate an un-aggregate by interaction of polyamines and long double stranded DNA chain [11, 12 and 13]. We believe that our finding on the variety morphology regulated structure in large long chain may also apply to other natural polymers chains, beside double stranded long DNA chains. From these contents, we could be constructed the channel switch like structural model regulated with polyamine-DNA self-assemble and disassemble process depend on the ratio balance of polyamines molecules. Further work of experimental

and theoretical simulation of DNA computation based on DNA channel switch model is in progress.

3.3 RNA Synthesis under aggregate conditions.

At RNA transcription, DNA aggregate at 2mM spermidine are observed on structural variation as shown Fig6. These conditions of transcription are carried out at the ATP: 0.5mM, Mg++:5mM, DNA: 48 µg/ml by RNA Synthesis Kit. In Spermidine-DNA aggregate, the synthesis of RNA's are observed these conditions.
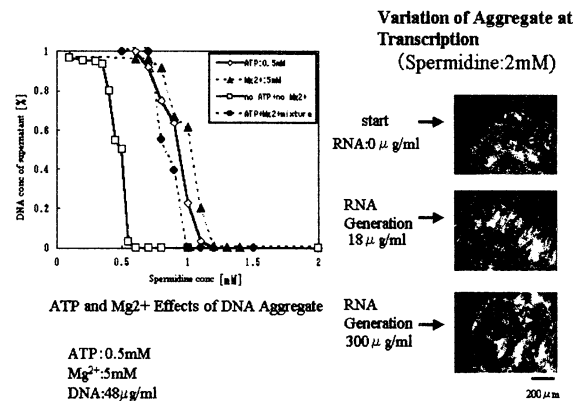
Figure 6:RNA Synthesis at various aggregate condition.

4. Structural Channel Switch Model based on experiment results

We are already presented on the Gene Expression on Liquid crystalling like Phenomena of Double stranded DNA chain as System model [11]. In this paper, we are more investigating the DNA aggregation and Dis-aggregation by existence of polyamines, ATP and Mg++.

From these results, we are proposed the structural channel switch model for Encode of DNA information in living organisms cells.
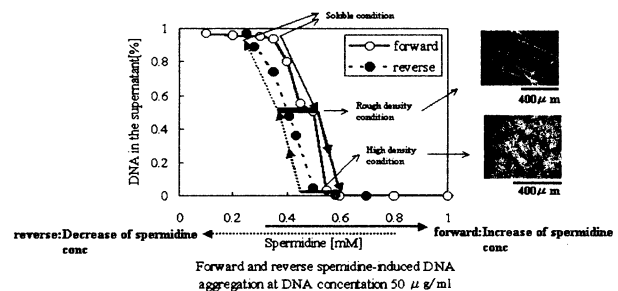
Figure 7:Control of aggregate form by Spermidine conc.

From these results(shown in Figure 7), We are presented the basic scheme as shown in Figure 8.

Namely, Long DNA chain in living cell are indicated the various formation depend on concentrations of polyamines, ATP and Mg++ as biological conditions. Dynamics at formation of Co-existence phase drived with Macro-aggregation and Micro-aggregation are control the region and it arrangement of RNA synthesis. We are investigate the more detail of long DNA chains processing Model based on Molecular Biophysical scheme in progress.
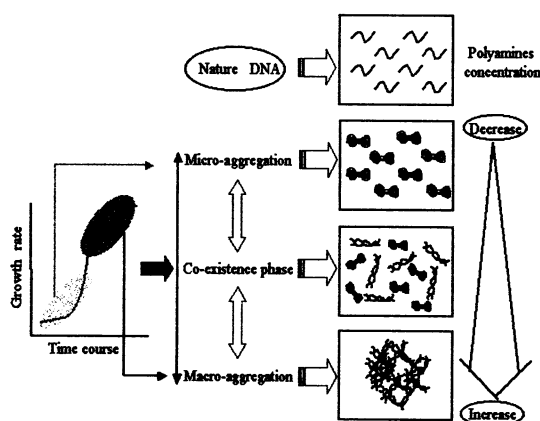
Figure-8: Control of Compaction and Relaxation for RNA Synthesis

Acknowledgements

References

[1] Cohen, S.S. (Ed): Introduction to the Polaymin, PreticeHill, EnglewwodCliffs, JJ, 1971

[2] Krasnow, MA, and Cozzarelli, N.R: Catenation of DNA ring by Topoisomerase. Mechanism of Control by Spermidine,Journal of Biol.Chem. 267, pp2687-2693. 1982

[3]Baeza,I.,Gariglio,P.,Rangel,L., Chavez,P.,Cervantes,L., Arguello,C., and Montanez, C:Electron Microscopy and Biochemical properties of Polyamine-Compacted DNA. Biochemistry 26, pp6387-6392 ,1987

[4] Sikorav, J.-L., and Livolant, F., A Liquid Crystalline Phase in Spermidine-Condensed DNA, Biophys.J. 67, pp1382-1392 ,1994

[5] Pelta,J.,Durand,D., Doucet,J.,and Livolant,F.,; DNA Mesophases induced by Spermidine:Structural Properties and Biological Implications, Biophys.J., 71, pp48-63, 1996

[6]Pelta,J.,Livolant,F and Sikorav,J.-L.,DNA aggregation induced polyamines and cobalthexamin., Journal of Biological Chemistry., 271, pp5656-5662. 1996

[7] Adelman, L.M., Molecular computation of solution to combinatorial problem, Science 266, pp1021-1023,1994

[8] Raspaud, E., Olvera de la Cruz, M., Sikorav, J.-L., and Livolant, F: Precipitation of DNA by polyamines: A polyelectrolyte behavior. Biophys.J., 74., pp381-393,1998

[9] Yonezawa, Y and Horikoshi, K., Polyamines in Alkalophilic Bacillus Y-25., Agriculture and Biological

Chemistry. VOl42,No10,pp1955-1965 ,1979

[10] Matsuzawa,Y., Yonezawa,Y and Yoshikawa,K., Formation of nucleation center in single double stranded DNA chain, Biochem.Biophy.Res.Comm, Vol 225,pp786-800 ,1996

[11] Yonezawa,Y and Kuramochi,H.,Gene Expression on Liquid crystalling like Phenomena of Double stranded DNA chain, Proceedings of the 1st International Conference of Systems Biology,pp 245-250,2000

[12] Kuramochi,H and Yonezawa,Y., Formation of Large Aggregates induced by Spermidine., Journal of BioScience and Bioengineering, Vol 92, No2,pp183-185, 2001

[13] Kuramochi,H and Yonezawa,Y., Formation in Liquid crystal like DNA aggregate under Biological condition: Phase Diagram, Morphology, and Biological Implication, Molecular Crystals and Liquid Crystals,Vol 367,pp487-495. 2001

[14] Yasuo Yonezawa ,Changing Phenomena from the coexistence phase of Macro & Micro-Aggregation by SpermidinemMg++ and ATP, Proceedings International Congress on Biological and Medical Engineering(in press), 2002.

Evolution from possible primitive tRNA-viroids to early poly-tRNA-derived mRNAs for synthesizing various house-keeping proteins

[1]Koji Ohnishi, [2]Madoka Ohshima and [3]Naotaka Furuichi .

[1,2]Dept .of Biology, Faculty of Science, and [3]Faculty of Agriculture,

Niigata University, Niigata, 950-2181, Japan.   [1]E-mail: ohnishi@sc.niigata-u.ac.jp

Abstract

Poly-tRNA theory have revealed that the *trrnD*-tRNA gene-cluster in the *Bacillus subtilis trrnD*-operon is a relic of early peptide synthesizing RNA apparatus. Futher analysis of poly-tRNA theory using recent databases elucidated that not only the genes of class II aminoacyl-tRNA synthetases (ARS) (GlyRS, AlaRS, AspRS, etc.) and F_1F_0-ATP-synthase subunits (F_0-a,F_1-gamma), but also C-type lectins, immunoglobulin V-domains (J-segment), RNA polymerase beta, and many other house-keeping protens were found to have been derived from a *trrnD*-mRNA*. Proto-tRNA ^{Gly} reconstructed from tRNA^{Gly} and other tRNAs was found to show base-sequence characteristics possibly capable of making a hammerhead-like structure closely similar to viroid hammer-heads. The results strongly suggest that proto-tRNA was most plausibly a viroid-like self-cleavable replicable ribozyme/ribo-organism.

1.   Introduction

What are the first living organisms like is an important unsolved problem in the studies of both biology and artificial life. RNA-first theory is now widely accepted [1-3, 3a], and tRNAs are considered to be ribozymes and/or descentants of early (intracellular?) ribo-organisms [4].

Recent studies of "poly-tRNA theory" (first proposed in 1993 by K.O.) [4-6] have revealed that tRNA gene clusters in the *Bacillus subtilis trrnD*- and *rrnB* gene clusters are almost undoubtedly relics of early peptide-synthesizing RNA-machines. They were RNA-machines for synthesizing 16- or 21-amino acid (aa)-long peptides, in which their aa-sequences are in the order of aa-specificities possessed by the linearly arranged tRNAs in the poly-RNA transcripts of the respective operons. In case of the *trrnD*-operon, a hypothetica l16-aa *trrnD*-peptide (Fig.1) would have been made on primitive ribosome consisting exclusively of 3 rRNAs encoded in the same operon. The tRNA^{Gly}-tRNA^{Cys}-tRNA^{Leu} region of this tRNA cluster is well conserved in almost every major evolutionary branch of contemporary eubacteria (examples are *trnGCL* clusters in *E. coli* and *Haemophilis influenzae*).

This paper reports that a reconstructed proto-tRNA^{Gly} is considered to be or to have had been capable of forming a hammer-head like structure [7], which strongly suggests that earliest tRNAs would have been viroid-like self-cleavable riboorganisms.

2. Evolution from a primitive tRNA^{Gly} to the *trrnD*-mRNA*

In the *trrnD*-type poly-tRNA-model (Fig. 1), first proposed in 1993, a hypothetical 16-aa "*trrnD*-peptide" is defined as a peptide whose aa-sequence is in the order of aa-specificities of the 16 tRNAs of the tRNA gene-cluster of the *B.subtilis trrnD*-operon. And a *trrnD*-mRNA is defined as an RNA complementary to the 16 anticodons as shown in Fig. 1). The theory hypothesizes that the *trrnD*-poly-tRNA (an RNA transcript of the tRNA gene-cluster region) would be an evolutionary relic of early RNA apparatus for synthesizing *trrnD*-peptide on primitive ribosome consisting of 3 rRNAs encoded by the same operon. This hypothesis can reasonably explain the genesis of eatliest mRNA (= *trrnD*-mRNA) by converting some RNA (most possibly, tRNA) to *trrnD*-mRNA (a most primitive mRNA) via selection of such base replacements that generate complementarities by anticodon triplets [4,6].

This model was further re-analyzed using recent GenBank and other databases. Proteins possessing aa-segments similar to the 16-aa *trrnD*-peptide (Fig.1) were searched from GenBank, EMBL, and SWISSPROT databases, using FASTA, BLAST, and psi-BLAST algorithms. As shown in Fig.1, 3-phosphoglycerate transporter protein B (pgtB protein) showed highest similarity to the *rrnB*-mRNA, giving a 64.4% (29/45) base-match and a matching probability by chance, $P_{nuc}(29,45)$ = 0.27×10^{-7} (See Pnuc(m,n) for ref.[4, 6]). Not only class II ARS (E. coli GlyRS alpha, AlaRS), and F_1F_0-ATP synthase subunits (F_0-a, F_1-gamma), as well as type-C lectins (lectin Bra-3, F_c-epsilon receptor, etc), RNA polymerase beta, and immunoglobulin V-domains (J-segments), were found to be homologues of the *trrnD*-mRNA-encoded proteins. Statistical evaluations have confirmed these homologies, which will be published elsewhere. Various house-keeping proteins of very different superfamilies were thus found to have had diversified from an earliest *trrnD*-mRNA (*trrnD*-peptide) in very early stage of organic evolution. It is also important to note that some of these genes (GlyS, F_0-a,F_1-gamma) are homologous to oligo-tRNA region (including tRNA^{Gly}) of the poly-tRNA.

Among tRNAs, tRNA^{Gly}(GGC) was found to be most similar to *trrnD*-mRNA. The *trrnD*-mRNA sequence was partly modified by introducing some wobble-pairing bases, as shown in Fig. 1., resulting in elucidating a more plausible earliest *trrnD*-type mRNA, here named "*trrnD*-mRNA*", as shown in Fig. 1. Base sequence comparison of

N S E V M D F T Y W H Q G C L L

5' CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA CCA 3'

trrnD-peptide

trrnD-poly-tRNA

3' UUG AGG CUU CAU UAC CUG AAG UGU AUG ACC GUG GUU CCG ACG AAU A AC 5'
anti-codons
*** *** *** *** *** *** *** *** *** *** *** *** *** *** * **

5' AAC UCC GAA GUA AUG GAC UUC ACA UAC UGG CAC CAA GGC UGC UUA U—UG 3'

trrnD-mRNA
(Hypothetical primitive mRNA, complementary to 16 anticodons)

5' AAU UCC GAA GUA AUG GAC UUC ACA UAC UGG CAC CAG GGU UGC UUA U—UG 3'
     G

trrnD-mRNA*
(includes Wobble-pairings)

Evolution from tRNA^Gly to *trrnD*-mRNA*

-3 ... 41   anticodon   tRNA^Gly *(Bacillus subtilis, trrnD)*
-3 ... 42   anticodon   tRNA^Gly *(Escherichia coli, trnGT)*
-2 ... 42   anticodon   tRNA^Ile *(Bacillus subtilis, trrnE)*
-2 ... 41   anticodon   tRNA^Trp *(Bacillus subtilis, trrnD)*

80 N S L V Q D F T W Q Q G T L L (D) 103
3-phosphoglycerate transporter protein B
(pgtB, Salmonella typhimurium)

130 G M E V T Q F T Y F Q Q V G G L 154
Glycyl-tRNA synthetase α
(GlyS, Escherichia coli)

G — E A T D F T Y W S S NN P
lectin Bra-3
(Hepatalanus rosa)

1 N W — — — — — F D Y W G Q G T M V 12
Immunoglobulin V-H domain
J-H 7 exon *(Xenopus laevis)*

106 N P — — — — — F D Y W G Q G V L V 117
V-H I, Sie *(Homo sapiens)*

38 E L E V G Q H F Y W Q I G — — — 40
Fo-ATP synthase a subunit
(Synechococcus sp., 6301)

Fig. 1. The *trrnD*-type poly-tRNA model of an early peptide-synthesizing poly-taRNA complex [4,5]. A primitive tRNA^Gly (= presumptive *trrnD*-mRNA) would habe been converted to *trrnD*-mRNA*, via interaction with and/or being manupilated by the 16 presumptive anticodons of the 16 tRNAs (= tRNA riboorganisms) of the *trrnD*-type poly-tRNA (= a cooperative tRNA complex). Base replacements (in the primitive tRNA^Gly) generating complementarities with anticodons would have been actively selected by the poly-tRNA, resulting in the emergence of an earliest *trrnD*-mRNA or *trrnD*-mRNA*. Amino acids and bases identical to the *trrnD*-peptide and the *trrnD*-mRNA are boxed, respectively. See text for detais.

[A] tRNA^Gly
(*B. subtilis trrnD*)

[B] *trrnD*-mRNA*

[C] *pgtB* DNA gene
(*Salmonella typhymurium*)
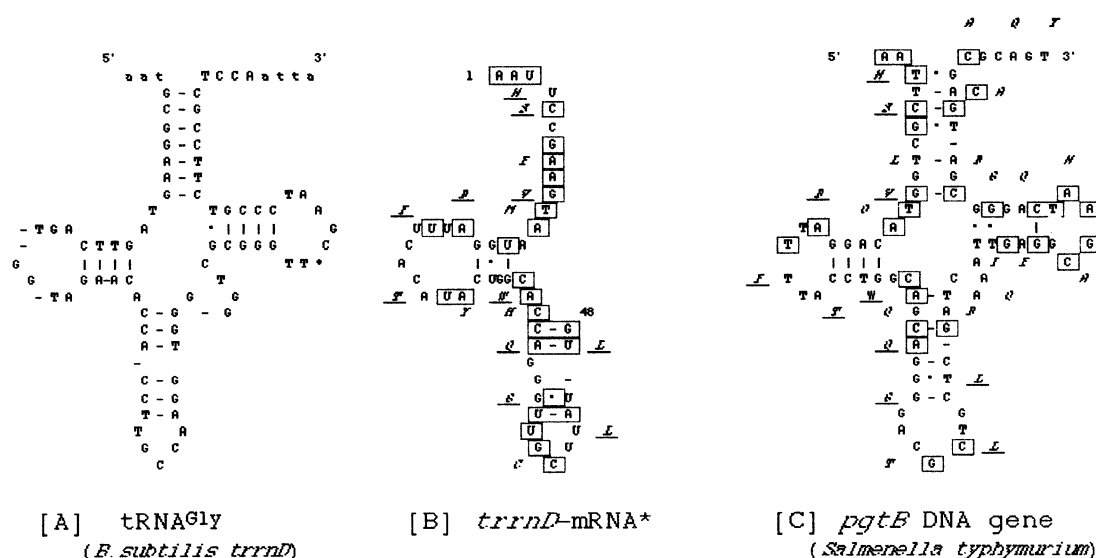
Fig. 2. Base sequences of the *trrnD*-mRNA* (B) and *pgtB* DNA gene (C) mapped onto the homologous base-positions of the cloverleaf model of the tRNA^Gly (A) in the *B. subtilis trrnD*-operon (poly-tRNA). Base matches to the tRNA^Gly are boxed. In [B] and [C], amino acids encoded by the base-sequences are shown in italisized capitals, where underlines indicate amino acid identity between (B) and (C). See [4].

the *trrnD*-mRNA* and the *pgtB* gene with tRNA^Gly (*trrnD*) on cloverleaf model elucidated that the base complementarities in the cloverleaf stems are evolutionarily very well conserved in the contemporary *pgtB*-mRNA (See Fig. 2[C].), strongly suggesting that contemporary *pgtB*-mRNA conserves structural relics of its ancestral / original tRNAs.

3. Origin of proto-tRNA^Gly as a viroid-like possibly replicable hammerhead ribozyme

A pseudo-tRNA gene cluster, recently found in the *E.coli rrnB* operon, consists of at least 6 pseudo-tRNAs, [4] and has a struture of pseudo-

NSEVMDFTYW, showing genuine homology to the deca-tRNA region of the *B. subtilis trrnD*-poly-tRNA region. The GCL region of the *B. subtilis trrnD*-poly-tRNA is closely homologous to the *trnGCL* tRNA-cluster in both *E. coli* and *H. influenzae*. The *GCL* tRNA cluster or its homologues are widely distributed in most major evolutionary branches of prokaryotes, as can be found by comparing complete genome databases in GenBank. It is thus easily concluded that the tRNA^Gly-tRNA^Cys-tRNA^Leu region of the *B.subtilis trrnD* tRNA-cluster are shared by most representative prokaryotic groups.

The base-sequence of proto-tRNA^Gly (GGC) was deduced from the alignment of the GCL-region of the *B. subtilis trrnD*-poly-tRNA with the *trnGCL* clusters and other tRNA genes, *pgtB*

Fig. 3.   A   proposed nucleotide sequence of proto-tRNA^{Gly}, deduced from an aligment of tRNAs,

viroids, and the *E. coli* 16S rRNA against the *trrnD*-mRNA and the *S. typhimurium pgtB* DNA gene.

Base-match levels of some comparisons aere evaluated by the level of matching probability by chance,

$P_{nuc}(m,n)$ value.   Abbreviations: ASBVd = avogado sunblotch viroid, CCCVd = coconut cadang-cadang

viroid, CEVd = *Citrus exocortis* viroid.   Sequence data in Figs. 1-4. were obtained from GenBank.

Viroid sequence data shown here are from   ref.[7].

gene, and the *trrnD*-mRNA(*), as shown in Fig. 3. Thus reconstructed sequence of the proto-tRNA was found to have a sequence possibly capable of making a hammerhead-like structure, as found in Fig. 4. Accordingly, some viroid sequences in the hammerhead-making region and its homologues were also aligned in Fig. 3.

As found in Fig. 4, most of the consensus hammerhead bases in (A) and (B) [7] are considerably well conserved in the proto-tRNA^{Gly} (C) and in the trrnD-mRNA (D),. Accordingly, early tRNAs such as proto-tRNA could have been

replicable riboorganism potentially capable of self-cleaving. Experimental data for determining whether or not artificially synthesized proto-tRNA (Fig. 4 (C)) could be active in self-cleaving would bring us with interesting information about early evolution of tRNA-ribozymes and early viroids.

(A) Consensus structure of Hammerhead Domain

(B) Hammerhead structure of ASBV viroid

(C) Hammerhead-like structure of proto-tRNA^{Gly}(GGC)

(D) Hammerhead-like structure of *trrnD*-mRNA together with its 5'-upstream region
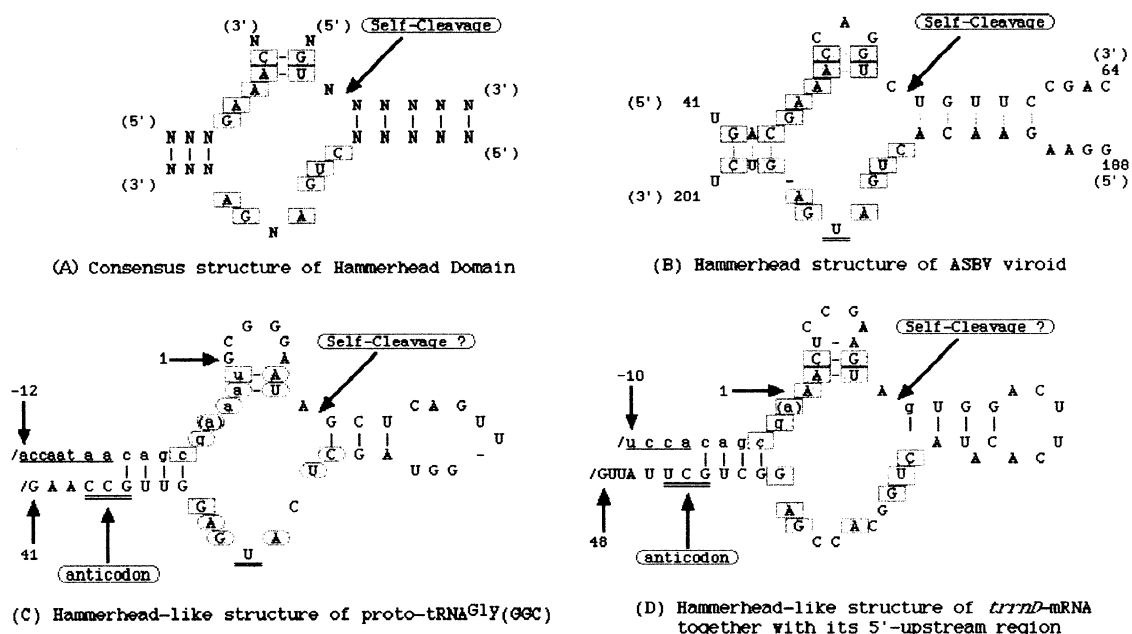
Fig. 4. Hammarhead structures in consensus domain and virusoid sequences (A,B) [7] and possible hammarhead-like structures in proto-tRNA and the *trrnD*-mRNA (C, D). Consensus bases of hammarhead structures are boxed. The proto-tRNA sequense was obtained in the comparative analysis in Fig. 3. lower-case letters indicate spacer region of the poly-tRNA, and the 5'-upstream region of the trrnD-mRNA. Arrows indicate either self-cleaving sits (in (A), and (B)), or possible self-cleaving site of proto-tRNA (C). The anticodon is indicated by an arrow in (C). See text for details.

References

[1] Haldane, JBS (1964), In: Fox, SW (ed), The Origins of Prebiological Systems and their Molecular Matrices, pp.11-15, Academic Press, New York.

[2] Eigen, M, Schuster, P (1979) The Hypercycle, Tspringer-Verlag, Berlin.

[3] Gesteland, R.F., Cech, T.R., Atkins, J.F. (1999), The RNA World, 2nd ed., Cold Spring Harbor Laboratory Press, Cold Spr.Harbor, NY.

[3a] Soll, D, Nishimura, S, Moore, PB (2001) "RNA", Pergamon, Amsterdam.

[4] Ohnishi, K (2002) Neural-network-like bio-machinogenesis and semeiogenesis. *Viva Origino* 30: 63-78..

[5] Ohnishi,K. (1993) Evolution from semi-tRNA to early protein-synthesizing RNA molecule. In: S.Sato et al._(ed) , "E"ndocyto-biology V", pp.407-414, Tubingen Univ.Press.

[6] Ohnishi,K. et al. (2002) Genome Informatics, 13: in press (10 pp.).

[7] Maramorosch, K (ed) (1991) "Viroids and Satellites", CRC Press, Boca Raton, Florida.

Computation of Electro kinetic Mobility of Charges Bio Molecules that Pass through the Ion Channel Pore on the Biological Excitable Membrane.

Hirohumi, Hirayama
Department of public Health Asahikawa medical college.
Higashi2-1, Midorigaoka, Asahikawa city 078, Japan.
hirayama@asahikawa-med.ac.jp

Abstract

The present paper introduce a method to compute the viscosity of mobility of a biomolecular particle passing through a channel pore on the excitable biological membrane. The equation of motion was described by usual newton's first law and the conservation law. The driving forces are composed of hydrodynamic, diffusion and electrical forces. For the simplicity, the electrical factor was set as constant value. We introduced a formula for viscosity of particle moving under the electrical filed surrounded by counter ions. The present paper will be available for predicting the passing time of bio molecular particle across the cellular membrane.

key words. Electro phoresis, kinetic mobility, Charges, Ion selective channel.

1. Introduction

Excitable cellular membrane has a lot of channels that pass a specific kind of ions such as calcium ion, sodium ions. They act as a filter to eliminate biochemical noises. The mechanism of movement of ions after the channel pore has open,however has still unknown. The main driving force may be hydro static, diffusion and electro motive force. The most characteristic factor is the electrical driving forces. The field potentials composed of the ions or passing particle itself, the fixed charges on the inner surface of the channel pore and their interaction including induced dipole moments. What is more interesting is the screening effect of counter ions around the particle. The description for mechanical feature of the movement of ion transport is thus, very complicated.

In the present work, we introduce a method for kinetic analysis of electro phoretic movement of solid spheres under the applied electrical field firstly proposed by Booth. The ultimate purpose of the present paper is to induce an analytic form of electro viscosity of the particle movement η.

Fig 1
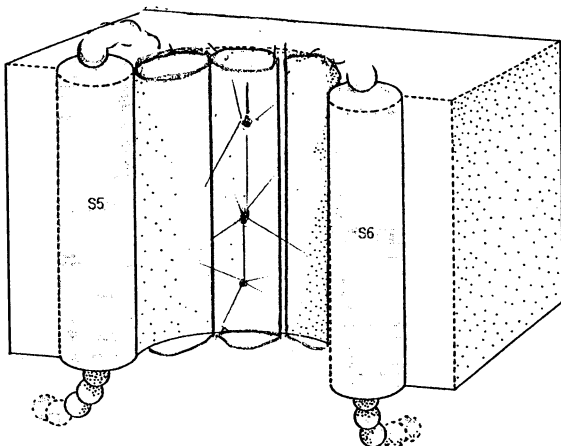
2. The basic assumptions.

We require two sets of equations, one for the motion of electrolyte and another of the motion of ions.

2-1. Electrolyte.

We neglected inertia and turbulence , slipping and assumed incompressible. Then

$$\nabla p + \rho \nabla \psi = \eta 0 \Delta v \quad \text{-------(2,1)}$$
$$\text{div } v = 0 \quad \text{-----(2,2)}$$

p is pressure, v is fluid velocity, ρ is the charge density and $\eta 0$ is the liquid viscosity, ψ is electrostatic potential. The Poisson equation connects ψ and ρ by

$$\Delta \psi = -4 \pi \rho / \varepsilon \quad \text{-----(2,3)}$$

ε is the di electric constant of the electrolyte.

We also assume that the volume of the solid is small compared to the total volume of the suspension

$$v <<< V. \text{-----(2,4)}$$

2-2. Ions.

The motion of ions are described by

$$ui = v - wi (kT/mi \nabla mi + e zi \nabla \psi) \quad \text{-----(2,5)}$$
$$\text{div } (ui \, mi) = 0 \quad \text{------(2,6)}$$

ui is the mean velocity of type i ion.
wi is mobility of type i ion,
mi is the concentration of type i ion.
zi is the valency of type i ion.

For solving equations (2,3) and (2,5), we separate

$$v = v^* + v^+ \quad \text{------(2,11)}$$
$$p = p^* + p^+ \quad \text{------(2,12)}$$
$$\psi = \psi 1 + \phi \quad \text{------(2,13)}$$

where v^* and p^* denote the solutions for v and p where there is no charge on the sphere. v^+ p^+ denote the disturbances in v and p evoked by presence of the charge on the particle.

3. Calculation of velocity distribution v*

Firstly, we transform the coordinates to spherical polar system by

$$x = r \sin\theta \cos\alpha, \quad y = r \sin\theta \sin\alpha \quad z = r \cos\theta$$

Then, the stress components are

$$S_{rr} = -p + 2\eta o\, f$$
$$S_{r\theta} = \eta o\, \partial f / \partial \theta$$
$$S_{r\alpha} = \eta o / \sin\theta\, \partial f / \partial \alpha \quad \text{-----(3,1)}$$

where f is a spherical harmonics of the second degree.

$$f(\theta,\alpha) = \alpha_{11} \sin^2\theta \cos^2\alpha + \alpha_{22} \sin^2\theta \sin^2\alpha$$
$$+ \alpha_{33} \cos^2\theta + 2\alpha_{23} \sin\theta \cos\theta \sin\alpha$$
$$+ 2\alpha_{13} \sin\theta \cos\theta \cos\alpha + 2\alpha_{12} \sin^2\theta \cos\alpha \sin\alpha$$

The r component of v have been found from Booth (1),

$$v_r^* = r\, f\, [\, 1 - 5a^3/(2r^3) + 3a^5/(2r^5)\,] \quad \text{--(3,3)}$$

The solution for the potential when the fluid is at rest can be expanded by a power series of charge on the spherical particle e Q

$$\psi_1 = \sum_{m=1} e^{2m-1} (\varepsilon a)^{-m} (kT)^{1-m} Q^m \lambda_m(\chi r) \text{-----(4,1)}$$

where $\lambda_1(x) = b/(1+b) \exp(b-x)/x$ ----(4,2)

$b = \chi a$ and

$$\chi^2 = 4e^2\pi \sum_{i=1} n_i z_i^2 /(\varepsilon kT)$$

n_i is the concentration of the type i ions at large distance from the particle,

e is the electronic charge.

5. Calculation of v and p.

Once we have solved for v* and ψ_1, we can have the general solution for velocity v.

The ionic concentration m_{1i} of type i ions when the fluid velocity is zero can be

$$m_{1i} = n_i \exp(-e z_i \psi_1/kT) \text{-----(5,1)}$$

We express

$$m_i = m_{1i} + s_i \quad \text{----(5,2)}$$

s_i is the distortion of the ionic concentrations by the fluid motion.

$$\Delta\phi = -4\pi e/\varepsilon \sum z_i s_i \quad \text{----(5,3)}$$

Multiplying (2,5) by m_i and taking divergence, we have

$$kT \Delta s_i + e z_i [\, s_i (\Delta\phi + \Delta\psi) + \partial\psi_1/\partial r\, \partial s_i/\partial r$$
$$+ \nabla s_i^* \nabla\phi + n_i \exp(-e z_i \psi_1/kT)$$
$$(\Delta\phi - e z_i/kT\, d\psi_1/dr\, \partial\phi/\partial r)\,]$$
$$= 1/w_i\, \text{div}\, v\, (\, n_i \exp(-e z_i \psi_1/kT) + s_i\,) \quad \text{----(5,4)}$$

We set the solutions that satisfy (5,4) in the series of powers of Q.

Qe is the total charge on the sphere when the system is at rest.

$$\phi = f(\theta,\alpha) \sum_{\nu=1} e^{2\nu-1} (\varepsilon a)^{-\nu} (kT)^{1-\nu} Q^\nu \phi_\nu(r) \quad \text{-----(5,5)}$$

$$s_i = f(\theta,\alpha) \sum_{\nu=1} e^{2\nu-1} (\varepsilon a)^{-\nu} (kT)^{1-\nu} Q^\nu \gamma_{i,\nu}(r) \quad \text{-----(5,6)}$$

$$v_r^* = f(\theta,\alpha) \sum_{\nu=1} e^{2\nu-2} \eta^{-1}o\, (\varepsilon a)^{1-\nu} (kT)^{2-\nu} Q^\nu \beta_{r,\nu}(r) \quad \text{----(5,7)}$$

$$v_\theta^* = \partial f/\partial\theta \sum_{\nu=1} e^{2\nu-2} \eta^{-1}o\, \varepsilon^{1-\nu} a^{1-\nu} (kT)^{2-\nu} Q^\nu \beta_{\theta,\nu}(r) \quad \text{-----(5,8)}$$

$$p^* = f \sum_{\nu=1} e^{2\nu-2} \varepsilon^{1-\nu} a^{2-\nu} (kT)^{2-\nu} Q^\nu \pi_\nu(r) \quad \text{-----(5,9)}$$

Substituting for ψ_1 from (4,1) and using (5,5) to (5,9) in (5,4),

$$\Delta_2^{(r)} (kT\, \gamma_{i,1} + e z_i n_i\, \phi_1)$$
$$= -n_i e z_i /(w_i kT)\, V(r)\, d\lambda_1/dr \quad (i = 1,2\text{--}s)$$

$$\Delta_2^{(r)} (kT\, \gamma_{i,2} + e z_i n_i\, \phi_2) =$$
$$-kT z_i (\gamma_{i,1} \Delta_0^{(r)} \lambda_1 + d\lambda_1/dr\, d\gamma_{i,1}/dr)$$
$$+ n_i e z_i^2 (\lambda_1 \Delta_2^{(r)} \phi_1 + d\lambda_1/dr\, d\phi_1/dr)$$
$$+ n_i e z_i V(r)/(w_i kT) (z_i \lambda_1\, d\lambda_1/dr + d\lambda_2/dr)$$
$$- n_i \varepsilon a z_i kT/(\eta o\, w_i e)\, d\lambda_1/dr\, \beta_{r,1}(r) \quad \text{----(5,10)}$$

where

$$\Delta_n^{(r)} = d^2/dr^2 + 2/r\, d/dr - n(n+1)/r^2 \quad \text{----(5,11)}$$

We separate

$$v_r^* = V(r) f(\theta,\alpha) \quad \text{------(5,12)}$$

from (3,3), is follows,

$$V(r) = r - 5a^3/(2r^2) + 3a^5/(2r^4) \text{----(5,13)}$$

The first step is to integrate (5,10),

$$kT\, \gamma_{i,1} + e z_i n_i\, \phi_1 = A_{i,1} r^2 + \beta_{i,1}(r)/r^3$$
$$+ n_i e z_i /(5 w_i kT) \int V(z)\, d\lambda_1(\chi z)/dz$$
$$z^2 (r^2/z^3 - z^2/r^3)\, dz \text{-----} (5,14)$$

From (4,2) and (5,12) we have

$$kT\, \gamma_{i,1} + e z_i n_i\, \phi_1 = B_{i,1}/r^3$$
$$+ n_i e z_i /(\chi^2 w_i kT)\, F_1(b, r\chi) \quad \text{--(5,15)}$$

where

$$F_1(b,x) = b\, e^{b-x} /(5(1+b))\, [F_{1a} + F_{1b} + F_{1c}] \text{--(5,16)}$$

$$F_{1a} = 5(1 + 3/x + 6/x^2 + 6/x^3)$$

$$F_{1b} = -5b^3/2\, (-x/8 + 1/8 - 1/(4x) + 3/(4x^2) + 2/x^3$$
$$+ x^2\, Ei(x)\, e^x/8)$$

$$F_{1c} = 3b^5/2\, (-x/144 + 1/144 - 1/(72x) + 1/(24x^2) - 1/(6x^2)$$
$$+ 5/(6x^2) + x^2\, Ei(x)\, e^x/144)$$

B_{i1} can be found from the boundary conditions on the surfae of the particle. The net for of each type of ion towards the surface of the particle vanishes at all the points of the surface. Denoting the radial component of the average ionic velocity

$$u_{ir}(r=a,\theta,\alpha) = 0 \text{ for all } \theta,\alpha \quad \text{---(5,17)}.$$

From (2,7) since vr =0 at r=a, (5,17) becomes

$$kT \, \partial \, mi/ \partial \, r \, (r=a) + mi \, e \, zi \, \partial \, \psi / \partial r \, (r=a) = 0$$

or from (5,5) and (5,6)

$$kT \, \partial \, \gamma_{i,1} / \partial \, r + mi \, e \, zi \, \partial \, \phi_1 / \partial \, r = 0 \, (r=a)$$

Applying this condition to 95,15), we have

$$B_{i,1} = ni \, e \, zi \, a^4 /(3 \, \chi \, wi \, kT) \, F1'(b,b) \quad --(5,19)$$

$$F1'(b,b) = d \, F(b,x)/dx \, (x = b) \quad ---(5,20)$$

From Poisson's relation

$$\Delta 2^{(r)} \, \phi_1 = - 4 \, \pi \, e/ \varepsilon \, \Sigma \, zi \, \gamma_{i,1} \quad ---(5,21)$$

Multiplying (5,15) by $4 \, \pi \, e \, zi /(\varepsilon \, kT)$ and summing
over i from i=1 to s,
we have a differential equation for ϕ_1 (r)

$$(\Delta 2^{(r)} - \chi^2) \, \phi_1 \, (r) = - 4 \, \pi \, e^2 /(\varepsilon \, \chi^2 \, k^2 \, T^2)$$
$$\sum_{i=1} ni \, zi^2 \, /wi \, (F1(b,r \chi) + \chi \, a^4 /(3 \, r^3)F1'(b,b)) \quad ---(5,22)$$

We require the solution of (2,1) and (2,2) where ψ is given
by

$$\psi = \psi_1 + \Psi \, f \quad ----(5,23)$$
Ψ is a function of r only. From Poisson relation

$$\rho = - \varepsilon /(4 \pi) (\Delta 0^{(r)} \, \psi_1 + f \Delta 2^{(r)} \, \Psi) ----(5,24)$$

Hence (2,1) becomes

$$\eta \, o \, curl \, curl \, v + \nabla p'$$
$$= \varepsilon /(4 \pi) [\, f \Delta 2^{(r)} \, \Psi \, \nabla \, \psi_1 + \Delta 0^{(r)} \, \psi_1 \nabla \, \Psi f]--(5,25)$$

$$p' = p + \varepsilon /(4 \pi) \int \Delta 0^{(r)} \, \psi_1 \, d \psi_1/dr \, dr$$

Substituting from (5,22), (5,24) for Ψ, we have the
viscosity η is in the form of
$$\eta = \eta \, o \, [\, 1 + 5 \, v /V \{ 1 + 2 \, e^2 \, Q^2 /(5 \eta \, o \, \varepsilon \, a^4)$$
$$(3 \, X1(a) + a \, dX1(r)/dr \, (r=a) - a^2 \, U1(a)) \, \}] --(6,10)$$

$$U1(r)=1/(20 \pi) \int d \lambda_1 (\chi \, z)/dz (\Delta 2^{(r)} - \chi^2) \, \phi_1 \, (r)*$$
$$*(- r^2/z^2 + z^3/r^3) \, dz \quad ----(6,11)$$

$$X1(r) = 1/5 \int (z \, dU1(z)/dz + 3 \, U1(z) (- r^2/z + z^4/r^3) \, dz$$
$$----(6,12)$$

Finally we have
$$\eta = \eta \, o \, [\, 1 + 5 \, v /(2V) \{ 1 + q^* (e^2 \, Q /(kT \, \varepsilon \, a))^2 \, Z(b) \}]$$
$$q^* = kT \, \varepsilon \, \Sigma \, ni \, zi^2 \, /wi \, / (\, \eta \, o \, e^2 \, \Sigma \, ni \, zi^2)$$
$$i=s$$

$$b = a \chi \quad --(6,15)$$
where
$$Z(b) = 1/(250 \, (1 +b)^2 \, \pi \,) \{ Z1 + Z2 + Z3 + Z4 + Z5*Z6)$$

$$Z1 = - 75/(4 \, b^2) - 25/(4 \, b) - 25/8 + 75b/8$$
$$- 2785 \, b^2/336 - 265b^3/168 - 1037b^4/224$$

$$Z2 = 593 \, b^5/672 - 67b^6/42 - b^7/14 + b^8/7$$

$$Z3 = e^{2b} \, E \, i(2b) (-25 \, b^2 + 25 \, b^5/3 + 10 \, b^7/3 - 2b^9/7)$$

$$Z4 = e^b \, E \, i(b) \, 75/16 \, (\, 1 - b^2 /30)(\, 2b^2 + 2 \, b^3 + b^4)$$

$$Z5 = F1'(b,b) \, (1 + b) \quad and$$

$$Z6 = Z6a + Z6b$$

$$Z6a = - 25 \, b/24 + 5b^2/8 - 5 \, b^3/12 + 35b^4/72$$
$$+ 5b^5 /288 - 5b^6/288$$

$$Z6b = -25/48 \, e^b \, Ei(b) \, b^5 \, (1 - b^2/30)$$

6. Computed results.

Fig 2 shows the computed results of Z(b) as a function
of b. $b = a \, \chi$. $1/\chi$ measures the thickness which is
normal to the surface of the ionic atmosphere around the
particle. b represents the ratio of the linear dimensions of
the particle to the double layer thickness. b >>1
corresponds to a thin double layer.
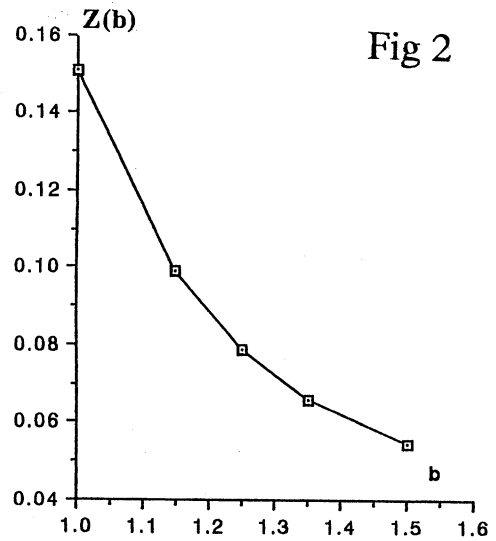
Fig 2

Fig 2 shows that as the ratio b increases contribution
from Z(b) on the electrical viscosity decreases rapidly.

7. Discussion.

We have introduced a method computing electro vis
in the suspensions of solid spherical particles. The
original approach has been proposed by Booth (1950).
Before discussing the computed results, we have to refer
for the assumptions particularly for the interface between
solid particle and surrounding circumstance electrolyte.
There must be required following assumptions on the
conditions at the solid-electrolyte interface.

1. The dielectric constant ε, the viscosity η and the
ionic mobility ωi are uniform through out the fluid
phase.
2. The electro phoretic velocity is proportional to the
magnitude of the applied field.
3. The inertia terms are neglected.
4. The electrolyte is symmetrical. It contains equal numbers
of ions of equal but opposite charge.
5. The thickness of the surface phase is small compared
to the particle radius. The surface phase refers to the
region that contain the surface charge on the particle.

6. The charge (ions) in the surface phase is immobile and there is no surface conductance. They can not move laterally over the surface. As a result, the surface charge density is fixed and being constant at any point . It remains constant when the electrolyte is set in motion.

7. The potential difference across the surface phase retains its equilibrium value when the external field is applied. Such potential difference is unaffected by the motion of the electrolyte.

8. The electrical double layer is small in thickness compared with the average distance between the neighboring particles. Since $1/\chi$ measures the thickness of the double layer around the particle,

$1/\chi$ << ro where

$4 \pi ro^3 N/3 = V$ and N is the total number of particles in the suspension. Since ro is set to be much larger than the radius if the solid sphere (r =a), from (2,4), the condition just above imposes no restriction on the relation of the double;e layer thickness to the size of the particle.

In the Fig 2, we computed the most interesting factor Z(b) as a function of b. Z(b) itself is not describe electroviscosity but occupies the main part. As b increases, the Z(b) value decreased rapidly. This means that the thickness of electrical double layer has become thin, the viscous influence becomes weak and the electrical interaction contributes little. As a result, the electro viscosity is decreased.

The present paper is just a starting point to predict the motion of bio molecules carrying charges. The problem should be strict when cylindrical circular boundary conditions have been imposed and taking into consideration of the electrical dipole moments produced by the inner surface of the polypeptide composing the pore.

References.

1. Booth F. " The cataphoresis of spherical solid non conducting particles in a symmetric electrolyte. Proc. Roy. Soc. London. vol 203. pp 514- . 1950.

2. Booth F. " The electro viscous effect for suspensions of solid spherical particles. Proc. Roy. Soc. London. vol 203. pp 533- . 1950.

3. Henry, D.C. Proc. Roy. Soc. A vol 133. 99196. 1931.

4. Cronwall.T.H Phys. Z. vol 28. pp 358. 1928.

5. Smoluchowski, M. Z. Phys. Chem. vol 93. pp 129. 1918.

6. Frohlich, H and Sack, R. proc. Roy. Soc.A, vol 185. pp 415. 1946.

Appendix

$\psi 1$ satisfies Poisson equation.

$$\Delta \psi 1 = -4 \pi e/\varepsilon \sum ni\, zi \exp(- e\, zi\, \psi 1/kT) \quad ----(A1)$$

The boundary conditions are

$$\psi 1 ------> 0, \quad r ----> oo. \quad ----(A2) \text{ and}$$
$$d\psi 1/dr = -e\, Q/(\varepsilon a^2) \quad r ---> a -----(A3)$$

where r is the distance from the center of the sphere. We expand the potential around the sphere by the series of charge Q such that.

$$\psi 1(x) = \sum \exp(2\nu -1) (\varepsilon a)^{(-\nu)} (- kT)^{(1-\nu)} Q^\nu \lambda_\nu(x)$$

where a is the radius of the particle and $x= \chi r$

The coefficient λ_n are calculated to

$\lambda 1(x) = b/(1+b) \exp(b-x)/x$
$\lambda 2(x)=0$
$\lambda 3(x) = -q3/6 (b/(1+b))^3 \exp(3b)/x*$
$*[\alpha(b) \exp(-x) - 2 \exp(x) Ei(4x) + \exp(-x) Ei(2x)]$
$\lambda 4(x)=0$ ---(12,a,b,c,d)

$$qm = \sum ni\, zi^{(m+1)} /\sum ni\, zi^2 \quad ---(13)$$
$$Ei*x) = \int_{x}^{oo} \exp(-u)/u\, du \quad --(14)$$

is the exponential integration.

$\alpha(b) = \exp(-2b)/(1+b) - Ei(2b)$
$\qquad + 2(1-b)/(1+b) \exp(2b) Ei(4b)$

II.Approximated solution for the velocity.

Henry calculated the electrophortic velocity on the assumptions that the potential due to the surface charge and the surrounding electrolyte preserve the spherical symmetry. From his work, we have for the first order approximation as

$U1 = \varepsilon E/(6\pi \eta) [\psi 1(b) (1+\sigma o)$
$+ 3 a^2 \sigma o (\int \psi 1(\chi r)/r^4 dr - 5 a^2 \int \psi 1(\chi r) /r^6 dr)]$
where
$\sigma o = (\sigma' - \sigma)/(2\sigma' +\sigma)$,
σ' is the specific conductivity of the electrolyte
σ is the specific conductivity of solid.

In the present case, $\sigma =0$, we have $\sigma o=1/2$. Substituting $\psi 1(x)$ for U1, we have the coefficients for the power series expansion

$c_{1,1} = E e/(6\pi \eta a) X1(b)$

$X1(b) = [1 + b^2/16 - 5 b^3 /48 - b^4/96 + b^5/96$
$\qquad + b^4 \exp(b) Ei(b) (1-b^2/12)/8]/(1+b)$
$c_{1,2} =0$
$c_{1,3} = E e^5 q3 X3(b)/(6\pi \eta a^3 (\varepsilon kT)^2$

$X3(b)a = \alpha(b) X1(b) (1+b)/b$
$X3(b)b = \exp(4b) Ei(4b) (-2/b -b/8 -5 b^2/24$
$\qquad + b^3/48 + b^4/48)$

$X3(b)c = -45 b^3 \exp(3b) Ei(3b) (1- 79/300 b^2)/32$

$X3(b)d = \exp(2b) Ei(2b) (1/b + b/16 - 5b^2/48$
$\qquad - b^3/96 + b^4/96)$

$X3(b)e = b^3(1- b^2/12)/8 (\int \exp(-x) Ei(2x)/x\, dx$
$\qquad - 2 \int \exp(x) Ei(4x)/x\, dx)$

$X3(b)f = -1/(48b) + 3/80 - 9 b /320 + 427 b^2 / 960$
$\qquad + 59 b^3 /1920 - 79 b^4/640$

$X3(b) = -(X3(b)a + X3(b)b + X3(b)c + X3(b)d$
$\qquad + X3(b)e + X3(b)f) (b/(1+b))^3)/6$

Characterization of Local Biophysical Electrical Conductivity and Viscosity of Multi Components Neural Transmitter System.

Hirohumi, Hirayama
Department of public Health Asahikawa medical college.
Higashi2-1, Midorigaoka, Asahikawa city 078, Japan.
hirayama@asahikawa-med.ac.jp

Abstract

Methods for calculating coefficients for mechanical diffusion, thermal diffusion, viscosity and thermal conductivity have been introduced by Chapman for the case of multi components biomoleuclar interaction system. The Boltzmann equation has been reduced by linear perturbation function with the help of equations of change. The integrals involved in the equations were reorganized by the Sonine polynomials which enabled the equation to analytic integration. These coefficients were expressed by matrix forms. The present method will be available for evaluating the integration of bio molecular collisional interactions.

key words. Boltzmann equation, Diffusion, Viscosity, Multi component system.

A. Introduction.

The ultimate mechanism of biomolecular particles is mechanical collision. Chapman et al proposed a lot of important works to describe the molecular collision and solved the integral differential equations for the distribution function $f_i(r,v,t)$. The present paper introduce a method to calculate coefficients of local physical field such as viscosity and diffusion.

Boltzmann equations can be expressed as
$$\partial f_i/\partial t + (v_i \cdot \partial f_i/\partial r) + 1/m_i(X_i \cdot f_i/\partial v_i) \quad ----(1)$$
$$= \Sigma J(f_i, f_j)$$
where f_i is a distribution fuction of particle velocity.
$$J(f_i, f_j) = \int \int \int (f'_i f'_j - f_i f_j) g_{ij} b \, db \, d\varepsilon \, dv_j ----(2)$$
$$f_i^{(1)}(r, v_i, t) = f_i^{(0)}(r, v_i, t) \, \phi_i(r, v_i, t) \quad ------(3-a)$$
ϕ_i is a preturbation function.
$$\phi_j = -(A_i \cdot \partial \log T/\partial r) - (B_j : \partial v_o/\partial r)$$
$$+ n \Sigma (C_j^{(k)} \cdot d_k) \quad ------(3-b)$$

In terms of the perturbation function, The Boltzmann integral differential equation can be expressed as
$$\partial f_i/\partial t + (v_i \cdot \partial f_i/\partial r) + 1/m_i(X_i \cdot f_i/\partial v_i)$$
$$= \Sigma \int \int \int f_i^{(0)} f_j^{(0)} (\phi_i' + \phi_j' - \phi_i - \phi_j) g_{ij} b$$
$$db \, d\varepsilon \, dv_j \quad ------(4)$$
With the help of equations of change in terms of ϕ_i
$$\partial n_i/\partial t + (\partial/\partial r \cdot n_i(v_o + V_i)) = 0 \quad --------(5)$$
$$\partial v_o/\partial t + v_o \cdot \partial v_o/\partial r = -1/\rho(\partial/\partial r \cdot p)$$
$$+ 1/\rho \Sigma n_i X_i \quad --------(6)$$
The resulting equation for the perturbation function is
$$f_i^{(0)} [n/n_i(V_i \cdot d_i) + (b_i : \partial v_o/\partial r)$$
$$- (5/2 - W_i^2)(V_i \cdot \partial \log T/\partial r)]$$
$$= \Sigma \int \int \int f_i^{(0)} f_j^{(0)} (\phi_i' + \phi_j' - \phi_i - \phi_j)$$
$$g_{ij} b \, db \, d\varepsilon \, dv_j \quad ----------(7)$$
The quantities d_i and b_i are defined by
$$d_i = \partial(n_i/n)/\partial r + (n_i/n - n_i m_i/\rho)\partial \log p/\partial r$$
$$- (n_i m_i/p\rho)[\rho/m_i X_i - \Sigma n_j X_j] \quad ------(8)$$
$$b_i = 2[W_i W_i - 1/3 W_i^2 U] \quad -------(9)$$
$$W_j = \sqrt{(m_i/2kT)} V_j$$

1. The coefficient of Diffusion.

The coefficient of diffusion in a multi component mixture can be obtained by taking only one term in the Sonine polynomial expansion of chapman's theory. The equations specifying the $c_{j,0}^{(h,k)}(1)$ are

$$\Sigma_j Q^{00}_{i,j} \, c_{j,0}^{(h,k)}(1) = - R^{(h,k)}_{i0} \quad -----(7,4-35)$$

These equations can be expressed in terms of $\Omega_{iL}^{(1,1)}$

$$\Sigma_j c_{j,0}^{(h,k)}(1) \Sigma_L n_L m_L /\{(m_i + m_L)\sqrt{m_j}\}$$
$$[n_i m_i(\delta_{ij} - \delta_{jL}) - n_j m_j(1 - \delta_{iL})] \Omega_{iL}^{(1,1)}$$
$$= -(\delta_{ih} - \delta_{ik})3/16\sqrt{(2kT)} \quad -----(7,4-36)$$

For a binary mixture, we set from (7,4-36) as

$$c_{i,0}^{(h,k)}(1) = (\delta_{ih} - \delta_{ik})3(m_i + m_j)/(16 n_i \rho)$$
$$\sqrt{(2kT/m_i)} 1/\Omega_{ij}^{(1,1)} \quad -----(7,4-37)$$

So that the first approximation to the coefficient of diffusion of a binary mixture is

$$D_{ij}(1) = 3(m_i + m_j)/(16 n m_i m_j) kT/\Omega_{ij}^{(1,1)}$$
$$-----(7,4-38)$$
From the definition, we have

$$c_{j,0}^{(h,k)}(1) = c_{j,0}^{(h,j)}(1) - c_{j,0}^{(k,j)}(1) \quad ---------(7,4-39)$$

Using this result, the expressions for the diffusion constants and the above relation for the binary diffusion constants, we get the equation for the general diffusion constants

$$\Sigma_j F_{i,j} \{ m_h D_{jh}(1) - m_k D_{jk}(1) = -(\delta_{ih} - \delta_{ik}) \quad -----(7,4-40)$$

$$F_{i,j} = 1/\rho \{ n_i/D_{ij}(1) + \Sigma_L n_L m_j/(m_i D_{iL}(1)) \}(1 - \delta_{ij})]$$
$$L \neq i \quad -------- (7,4-41)$$

$F^{ij} = (-1)^{i+j}$ I I

Solving (7,4-40), we get

m_h $D_{ih}(1)$ - m_k $D_{ik}(1) = (F^{hi} - F^{ki}) / IFI$ ---(7,4-43)

$D_{ij}(1) = (F^{ji} - F^{ii}) / (m_j IFI)$ ------(7,4-44)

m_h $D_{ih}(1)$ - m_k $D_{ik}(1) = (F^{hi} - F^{ki}) / IFI$ ---(7,4-43)

$D_{ij}(1) = (F^{ji} - F^{ii}) / (m_j IFI)$ ------(7,4-44)

2. The coefficient of thermal diffusion.

To obtain the thermal diffusion constant, two terms in the Sonine polynomial expansion should be considered. Since

$$\sum_j \sum_{m'=0}^{1} Q^{m m'}{}_{i,j} \; a_{j,m'}(2) = - R_{im} \quad -----(7,4-49)$$

$Q^{m m'}{}_{i,j}$ can be expressed in terms of $\Omega_{ik}^{(1,1)}$ as

$Q^{0\,0}{}_{i,j} = 8 \sum_k n_k m_k / \{ (m_i + m_k) \sqrt{(m_i m_j)} \}$

$[n_i m_i (\delta_{ij} - \delta_{jk}) - n_j m_j (1 - \delta_{ik})] \Omega_{ik}^{(1,1)}$
$\qquad\qquad\qquad\qquad -----(7,4-50)$

$Q^{0\,1}{}_{i,j} = - 8 (m_i/m_j)^{3/2} \sum_k n_i n_k m_k^2 / \{ (m_i + m_k)^2 \}$

$(\delta_{ij} - \delta_{kj}) [\Omega_{ik}^{(1,2)} - 5/2 \Omega_{ik}^{(1,1)}] \quad -----(7,4-51)$

$Q^{1\,0}{}_{i,j} = m_j / m_i \; \Omega_{ij}^{(0,1)} \qquad\qquad -----(7,4-52)$

$Q^{1\,1}{}_{i,j} = 8 (m_i/m_j)^{3/2} \sum_k n_i n_k m_k / \{ (m_i + m_k)^3 \} *$

$*[A + B]$
$A = (\delta_{ij} - \delta_{kj}) (A1 + A2 + A3)$
$A1 = 5/4 (6 m_j^2 + 5 m_k^2) \Omega_{ik}^{(1,1)}$
$A2 = - 5 m_k^2 \Omega_{ik}^{(1,2)}$
$A3 = m_k^2 \Omega_{ik}^{(1,3)}$
$B = (\delta_{ij} + \delta_{jk}) 2 m_j m_k \Omega_{ik}^{(2,2)} \qquad -----(7,4-53)$
$R_{im} = \delta_{mi} 15/4 n_i \sqrt{(2 kT/m_i)} \quad --- (7.4-54)$

The expressions in (7,4-49) constitutes a set of linear equations for the coefficients a_{j0} and a_{j1}. They can be solved by Cramer's method. Then the coefficient of thermal diffusion can be

$D_i^T(2) = n_i \sqrt{(m_i kT/2)} IA I / IBI$

3. The coefficient of viscosity

For a ν component mixture the first approximation to the viscosity is

$$\eta (1) = kT/2 \sum_j n_j b_{j0}(1) \quad ----(7,4-56)$$

$b_{j0}(1)$ are then determined by the ν equations.

$$\sum_j (Q^{00}{}_{i,j} / R_{i,j}) b_{j0}(1) = -1, \quad i = 1,2 --- \nu \; --(7,4-57)$$

$Q^{00}{}_{i,j} = \sum_L n_i n_L \{ \delta_{i,j}[W_i ; W_i]_{i,L} + \delta_{j,L} [W_i ; W_L]_{i,L} \}$

where
$W_i = W_i W_i - 1/3 W_i^2 U \quad ----(7,4-59)$

$R_{i,0} = \int 2 (W_i ; W_i) f_i^{(0)} dV_i$
$= -4/3 \int W_i^4 f_i^{(0)} dV_i$
$= - 5 n_i \qquad\qquad -----(7,4-60)$

from (7,4-57), we can get the $b_{j0}(1)$ as ratios of two determinants of order ν by Cramer's method. The only quantity that manifests in the expression for the coefficient of viscosity is $\sum n_j b_{j0}(1)$. This can be written as a ratio of two determinants. The one in the numerator of order ($\nu +1$) and that in the denominator of order ν. We get

$$\eta (1) = - IHI / IH_{ij}I \quad ----(7,4-61)$$

$H_{ij} = - 2 n_i Q^{00}{}_{i,j} / (n^2 k T R_{i,0})$
$= 2/5 \; 1/(kT) *$
$* \sum_L n_i n_L /n^2 \{ \delta_{i,j}[W_i ; W_i]_{i,L} + \delta_{j,L} [W_i ; W_L]_{i,L} \}$
$\qquad\qquad ----(7,4-62)$
or
$H_{i,j} = 32/15 (n_i m_i/kT n^2 m_j)$
$\sum_L n_L m_L / \{ (m_i + m_L)^2 \} *[A + B]$
$A = 5 m_j (\delta_{ij} - \delta_{jL}) \Omega_{iL}^{(1,1)}$
$B = 3/2 m_L (\delta_{ij} + \delta_{jL}) \Omega_{iL}^{(2,2)} \quad ----(7,4-63)$

4 The coefficient of Thermal conductivity

The expression for the energy flux is expressed in terms of the diffusion velocities and the temperature gradient.

$q = - \lambda \; \partial T / \partial r + 5/2 kT \sum_i n_i V_i$

$+ kT/n \sum_i \sum_j n_j D_{ij}^T/(m_i D_{ij}(1)) (V_i - V_j) ---(7,4-64)$

$\lambda = \lambda '(2) - k/(2 n) \sum_i \sum_j n_j n_i / D_{ij}(1) *$

$*[D_i^T/(n_i m_i) - D_j^T/(n_j m_j)]^2 ---(7,4-65)$
λ is the usual coefficient of thermal conductivity.
and
$\lambda '(2) = -75/8 k^2 T \; IAA I/IBBI$

B. Computed results.
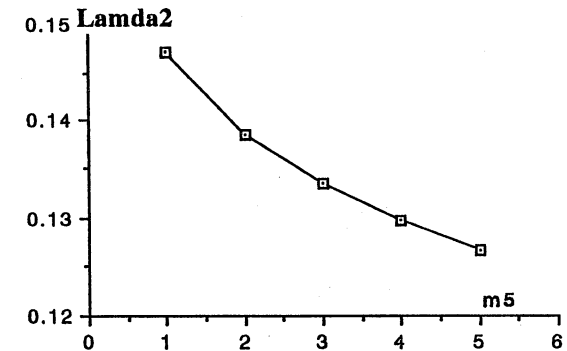
Fig shows influence of changes in mass m5 on the thermal conductivity for five components system. Deep discussion will be required to compare with biological experimental data.

Reference
1. Molecular theory of gas and liquid. ed by Hirshfelder and Curtis. 1968. chap 7.

Appendix I.

a. Diffusion and thermal diffusion coefficients in terms of Sonine expansion coefficients.

The integral for diffusion velocity can be expressed by

$$\underline{V_i} = 1/n_i \int V_i\, f_i\ dV_i$$

Using the perturbation function of ϕ_1 for distribution function and the terms involving B_i vanishes on integration,

$$\underline{V_i} = 1/n_i \int \{ n \sum (C_i^{(j)} \cdot d_j) - (A_i \cdot \partial \log T / \partial r) \}^*$$
$$*V_i\, f_i^{(0)}\ dV_i$$

using the form of the function A_i and $C_i^{(j)}$,

$$\underline{V_i} = n^2/(n_i\, \rho) \sum m_j\, D_{ij}\, d_j - 1/(n_i\, m_i) D_i^T\, \partial \log T / \partial r)$$

where

$$D_{ij} = \rho /(3\, n\, m_i) \sqrt{(2\, kT/m_i)} \int (C_i^{(j)})(W_i)\ W_i^2\ f_i^{(0)}\ dV_i$$
$$D_i^T = m_i/3 \sqrt{(2\, kT/m_i)} \int (A_i(W_i)\ W_i^2\ f_i^{(0)}\ dV_{ij}$$

These are diffusion coefficients and thermal diffusion coefficients for multi component. The Sonine polynomial expansion reduces them

$$D_{ij}(\xi) = \rho /(3\, n\, m_i) \sqrt{(2\, kT/m_i)} \sum c_{im}^{(j,i)}(\xi)$$
$$\int (S_{3/2}^{(m)}(W_i^2))\ f_i^{(0)}\ dV_i$$

$$D_i^T(\xi) = m_i/3 \sqrt{(m_i/2kT)} \sum a_{im}(\xi)$$
$$\int (S_{3/2}^{(m)}(W_i^2))\ V_i^2\ f_i^{(0)}\ dV_i$$

b. Coefficient of viscosity in terms of Sonine expansion coefficients

The integral for the pressure tensor in terms of the perturbation function, ϕ_1

$$p = \sum m_j \int V_j\, V_j\ f_j\ dV_j$$
$$= \sum m_j \{ \int V_j\, V_j\ f_j^{(0)}\ dV_j + \int V_j\, V_j\ f_j^{(0)}\ \phi_j\ dV_j \}$$
$$= pU + \sum m_j \int V_j\, V_j\ f_j^{(0)}\ \phi_j\ dV_j$$

where

$p = nkT$ is the equilibrium hydrostatic pressure at the local temperature and density. using ϕ_1

$$= pU - \sum m_j \int V_j\, V_j\ f_j^{(0)} (\ B_j : \partial v_0 / \partial r)\ dV_j$$

$$p = pU - \{ 2/15 \sum m_j^2 /(2\, kT) \int (B_j(W_i)\ V_j^4\ f_j^{(0)}\ dV_j \}\, S$$

S is the rate of shear tensor defined by

$$S_{\alpha\beta} = 1/2 [\partial v_{0\beta}/\partial x_\alpha + \partial v_{0\alpha}/\partial x_\beta] - 1/3\, \delta_{\alpha\beta} (\partial / \partial r \cdot v_0)$$

The coefficient of viscosity η is defined by the relation

$$p = pU - 2\, \eta\, S$$

Then, we have

$$\eta = 1/15 \sum m_j^2 /(2\, kT) \int (B_j(W_j)\ V_j^4\ f_j^{(0)}\ dV_j$$

Using the Sonine polynomial

$$\eta = 1/15 \sum m_j^2 /(2\, kT) \sum \int S_{5/2}^{(m)}(W_j^2)\ V_j^4\ f_j^{(0)}\ dV_j$$

c. Coefficient of thermal conductivity in terms of Sonine expansion coefficients.

The integral for energy flux vector q is given in terms of the perturbation function, ϕ_1

$$\mathbf{q} = 1/2 \sum m_j \int V_j^2\ V_j\ f_j^{(0)}\ \phi_1\, dV_j$$

Substituting ϕ_1 for this expression, the term containing B_j does not contribute,

$$= 1/2 \sum m_j \int \{ n \sum (C_j^{(k)} \cdot d_k) - (A_j \cdot \partial \log T / \partial r) \}$$
$$V_j^2\ V_j\ f_j^{(0)}\ dV_j$$

The energy flux can be separated into two parts,
1. Due to the flux of molecules relative to the mass velocity and
2. other causes. Then

$$= 5/2\, kT \sum n_j\, \underline{V_j}$$
$$- kT \sum \int \{ n \sum (C_j^{(k)}(W_j) (\ W_j \cdot d_k) - A_j(W_j)$$
$$(\ W_j \cdot \partial \log T / \partial r) \} (\ 5/2 - W_j^2\ )\ V_j\ f_j^{(0)}\ dV_j$$

$$= 5/2\, kT \sum n_j\, \underline{V_j}$$
$$- \lambda'\, \partial T / \partial r$$
$$- \sqrt{2/3}\, n\, (kT)^{3/2} \sum d_k \sum 1/\sqrt{m_j}$$
$$\int (C_j^{(k)}(W_j) (\ 5/2 - W_j^2\ )\ W_j^2\ f_j^{(0)}\ dV_j$$

$$\lambda' = - \sqrt{2/3}\, k\, (kT)^{1/2} \sum 1/\sqrt{m_j}$$
$$\int (A_j(W_i) (\ 5/2 - W_j^2\ )\ W_j^2\ f_j^{(0)}\ dV_j$$

$$\lambda'(\xi) = - \sqrt{2/3}\, k\, (kT)^{1/2} \sum 1/\sqrt{m_j}\, a_{jm}(\xi)$$
$$\int S_{3/2}^{(m)}(W_j^2) (\ 5/2 - W_j^2\ )\ W_j^2\ f_j^{(0)}\ dV_j$$

where

$$\Omega_{ij(L,s)} = \sqrt{(2\pi kT/\mu_{ij})} \int \int \exp(-\gamma_{ij}^2)\ \gamma_{ij}^{(2s+3)}$$
$$(\ 1 - \cos^L \chi\ )\, b\ db\ d\gamma_{ij}$$

Appendix II.

$$F^{ij} = (-1)^{i+j}\ I\ I$$

I I =	F$_{11}$	F$_{12}$	F$_{13}$	------	F$_{1,j-1}$	F$_{1,j+1}$	F$_{1,j+2}$	F$_{1,j+3}$	-------------	F$_{1,\nu-1}$	F$_{1,\nu}$
	F$_{21}$	F$_{22}$	F$_{23}$	------	F$_{2,j-1}$	F$_{2,j+1}$	F$_{2,j+2}$	F$_{2,j+3}$	-------------	F$_{2,\nu-1}$	F$_{2,\nu}$
	F$_{31}$	F$_{32}$	F$_{33}$	------	F$_{3,j-1}$	F$_{3,j+1}$	F$_{3,j+2}$	F$_{3,j+3}$	-------------	F$_{3,\nu-1}$	F$_{3,\nu}$
	F$_{i-2,1}$	F$_{i-2,2}$	F$_{i-2,3}$	--	F$_{i-2,j-1}$	F$_{i-2,j+1}$	F$_{i-2,j+2}$	F$_{i-2,j+3}$	-------	F$_{i-2,\nu-1}$	F$_{i-2,\nu}$
	F$_{i-1,1}$	F$_{i-1,2}$	F$_{i-1,3}$	--	F$_{i-1,j-1}$	F$_{i-1,j+1}$	F$_{i-1,j+2}$	F$_{i-1,j+3}$	-------	F$_{i-1,\nu-1}$	F$_{i-1,\nu}$
	F$_{i+1,1}$	F$_{i+1,2}$	F$_{i+1,3}$	--	F$_{i+1,j-1}$	F$_{i+1,j+1}$	F$_{i+1,j+2}$	F$_{i+1,j+3}$	----	F$_{i+1,\nu-1}$	F$_{i+1,\nu}$
	F$_{i+2,1}$	F$_{i+2,2}$	F$_{i+2,3}$	--	F$_{i+2,j-1}$	F$_{i+2,j+1}$	F$_{i+2,j+2}$		F$_{i+2,j+3}$	---- F$_{i+2,\nu-1}$	F$_{i+2,\nu}$
	F$_{\nu,1}$	F$_{\nu,2}$	F$_{\nu,3}$	--	F$_{\nu,j-1}$	F$_{\nu,j+1}$	F$_{\nu,j+2}$	F$_{\nu,j+3}$	----	F$_{\nu,\nu-1}$	F$_{\nu,\nu}$

$IAI =$

$$
\begin{array}{ccccccccc}
Q^{00}_{1,1} & Q^{00}_{1,2} & Q^{00}_{1,j} & Q^{00}_{1,\nu} & & Q^{01}_{1,1} & Q^{01}_{1,2} & Q^{01}_{1,j} & Q^{01}_{1,\nu} & 0 \\
Q^{00}_{2,1} & Q^{00}_{2,2} & Q^{00}_{2,j} & Q^{00}_{2,\nu} & & Q^{01}_{2,1} & Q^{01}_{2,2} & Q^{01}_{2,j} & Q^{01}_{2,\nu} & 0 \\
Q^{00}_{i,1} & Q^{00}_{i,2} & Q^{00}_{i,j} & Q^{00}_{i,\nu} & & Q^{01}_{i,1} & Q^{01}_{i,2} & Q^{01}_{i,j} & Q^{01}_{i,\nu} & 0 \\
& & & & & & & & & \\
Q^{00}_{\nu,1} & Q^{00}_{\nu,2} & Q^{00}_{\nu,j} & Q^{00}_{\nu,\nu} & & Q^{01}_{\nu,1} & Q^{01}_{\nu,2} & Q^{01}_{\nu,j} & Q^{01}_{\nu,\nu} & 0 \\
& & & & & & & & & \\
Q^{10}_{1,1} & Q^{10}_{1,2} & Q^{10}_{1,j} & Q^{10}_{1,\nu} & & Q^{11}_{1,1} & Q^{11}_{1,2} & Q^{11}_{1,j} & Q^{11}_{1,\nu} & R_{11} \\
Q^{10}_{2,1} & Q^{10}_{2,2} & Q^{10}_{2,j} & Q^{10}_{2,\nu} & & Q^{11}_{2,1} & Q^{11}_{2,2} & Q^{11}_{2,j} & Q^{11}_{2,\nu} & R_{21} \\
Q^{10}_{i,1} & Q^{10}_{i,2} & Q^{10}_{i,j} & Q^{10}_{i,\nu} & & Q^{11}_{i,1} & Q^{11}_{i,2} & Q^{11}_{i,j} & Q^{11}_{i,\nu} & R_{i1} \\
& & & & & & & & & \\
Q^{10}_{\nu,1} & Q^{10}_{\nu,2} & Q^{10}_{\nu,j} & Q^{10}_{\nu,\nu} & & Q^{11}_{\nu,1} & Q^{11}_{\nu,2} & Q^{11}_{\nu,j} & Q^{11}_{\nu,\nu} & R_{\nu}1 \\
\delta_{i,1} & \delta_{i,2} & \delta_{i,j} & \delta_{i,\nu} & & 0 & 0 & 0 & 0 & 0 \\
\end{array}
$$

$IBI =$

$$
\begin{array}{cccccccc}
Q^{00}_{1,1} & Q^{00}_{1,2} & Q^{00}_{1,j} & Q^{00}_{1,\nu} & & Q^{01}_{1,1} & Q^{01}_{1,2} & Q^{01}_{1,j} & Q^{01}_{1,\nu} \\
** & ** & ** & ** & ** & ** & & & \\
Q^{00}_{\nu,1} & Q^{00}_{\nu,2} & Q^{00}_{\nu,j} & Q^{00}_{y,\nu} & & Q^{01}_{\nu,1} & Q^{01}_{\nu,2} & Q^{01}_{\nu,j} & Q^{01}_{\nu,\nu} \\
Q^{10}_{1,1} & Q^{10}_{1,2} & Q^{10}_{1,j} & Q^{10}_{1,\nu} & & Q^{11}_{1,1} & Q^{11}_{1,2} & Q^{11}_{1,j} & Q^{11}_{1,\nu} \\
Q^{10}_{i,1} & Q^{10}_{i,2} & Q^{10}_{i,j} & Q^{10}_{i,\nu} & & Q^{11}_{i,1} & Q^{11}_{i,2} & Q^{11}_{i,j} & Q^{11}_{i,\nu} \\
& & & & & & & & \\
Q^{10}_{\nu,1} & Q^{10}_{\nu,2} & Q^{10}_{\nu,j} & Q^{10}_{\nu,\nu} & & Q^{11}_{\nu,1} & Q^{11}_{\nu,2} & Q^{11}_{\nu,j} & Q^{11}_{\nu,\nu} \\
\end{array}
$$

$iAAI =$

$$
\begin{array}{ccccccccc}
q^{00}_{1,1} & q^{00}_{1,2} & q^{00}_{1,j} & q^{00}_{1,\nu} & & q^{01}_{1,1} & q^{01}_{1,2} & q^{01}_{1,j} & q^{01}_{1,\nu} & 0 \\
q^{00}_{2,1} & q^{00}_{2,2} & q^{00}_{2,j} & q^{00}_{2,\nu} & & q^{01}_{2,1} & q^{01}_{2,2} & q^{01}_{2,j} & q^{01}_{2,\nu} & 0 \\
q^{00}_{i,1} & q^{00}_{i,2} & q^{00}_{i,j} & q^{00}_{i,\nu} & & q^{01}_{i,1} & q^{01}_{i,2} & q^{01}_{i,j} & q^{01}_{i,\nu} & 0 \\
& & & & & & & & & \\
q^{00}_{\nu,1} & q^{00}_{\nu,2} & q^{00}_{\nu,j} & q^{00}_{\nu,\nu} & & q^{01}_{\nu,1} & q^{01}_{\nu,2} & q^{01}_{\nu,j} & q^{01}_{\nu,\nu} & 0 \\
& & & & & & & & & \\
q^{10}_{1,1} & q^{10}_{1,2} & q^{10}_{1,j} & q^{10}_{1,\nu} & & q^{11}_{1,1} & q^{11}_{1,2} & q^{11}_{1,j} & q^{11}_{1,\nu} & 1 \\
q^{10}_{2,1} & q^{10}_{2,2} & q^{10}_{2,j} & q^{10}_{2,\nu} & & q^{11}_{2,1} & q^{11}_{2,2} & q^{11}_{2,j} & q^{11}_{2,\nu} & 1 \\
q^{10}_{i,1} & q^{10}_{i,2} & q^{10}_{i,j} & q^{10}_{i,\nu} & & q^{11}_{i,1} & q^{11}_{i,2} & q^{11}_{i,j} & q^{11}_{i,\nu} & 1 \\
& & & & & & & & & \\
q^{10}_{\nu,1} & q^{10}_{\nu,2} & q^{10}_{\nu,j} & q^{10}_{\nu,\nu} & & q^{11}_{\nu,1} & q^{11}_{\nu,2} & q^{11}_{\nu,j} & q^{11}_{\nu,\nu} & 1 \\
0 & 0 & 0 & 1 & & 1 & 1 & 1 & 1 & 0 \\
\end{array}
$$

$IBBI =$

$$
\begin{array}{cccccccc}
q^{00}_{1,1} & q^{00}_{1,2} & q^{00}_{1,j} & q^{00}_{1,\nu} & & q^{01}_{1,1} & q^{01}_{1,2} & q^{01}_{1,j} & q^{01}_{1,\nu} \\
** & ** & ** & ** & ** & ** & & & \\
q^{00}_{\nu,1} & q^{00}_{\nu,2} & q^{00}_{\nu,j} & q^{00}_{\nu,\nu} & & q^{01}_{\nu,1} & q^{01}_{\nu,2} & q^{01}_{\nu,j} & q^{01}_{\nu,\nu} \\
q^{10}_{1,1} & q^{10}_{1,2} & q^{10}_{1,j} & q^{10}_{1,\nu} & & q^{11}_{1,1} & q^{11}_{1,2} & q^{11}_{1,j} & q^{11}_{1,\nu} \\
q^{10}_{i,1} & q^{10}_{i,2} & q^{10}_{i,j} & q^{10}_{i,\nu} & & q^{11}_{i,1} & q^{11}_{i,2} & q^{11}_{i,j} & q^{11}_{i,\nu} \\
& & & & & & & & \\
q^{10}_{\nu,1} & q^{10}_{\nu,2} & q^{10}_{\nu,j} & q^{10}_{\nu,\nu} & & q^{11}_{\nu,1} & q^{11}_{\nu,2} & q^{11}_{\nu,j} & q^{11}_{\nu,\nu} \\
\end{array}
$$

$$q^{mm'}_{i,j} = \sqrt{(m_i\, m_j)/(n_i\, n_j)}\; Q^{mm'}_{i,j}$$

$$
IHI =
\begin{array}{ccccc}
H_{1,1} & H_{1,2} & H_{1,j} & H_{1,\nu} & n_1/n \\
H_{2,1} & H_{2,2} & H_{2,j} & H_{2,\nu} & n_2/n \\
H_{i,1} & H_{i,2} & H_{i,j} & H_{i,\nu} & n_i/n \\
& & & & \\
H_{\nu,1} & H_{\nu,2} & H_{\nu,j} & H_{\nu,\nu} & n_{\nu}/n \\
n_1/n & n_2/n & n_j/n & n_{\nu}/n & 0 \\
\end{array}
$$

Automatic Evaluation of EEG Recording Based on Artificial Intelligence of Electroencephalographers

M.NAKAMURA, ○ Q.CHEN, T.SUGI
Dept. of Advanced Systems Control
Engineering, Saga University
Honjomachi, Saga 840-8502, Japan
E-mail:chenqian@cntl.ee.saga-u.ac.jp

H.SHIBASAKI
Dept. of Neurology and Human Brain
Research Center, Kyoto University
Yoshida-Honmachi, Kyoto 606-8501, Japan

Abstract

EEG recording is important no matter in visual inspection or in the computer-assisted automatic EEG interpretation system. In order to acquire qualified EEG record, we proposed a method to estimate the status of the subject during recording; and evaluate the quality of the records at the same time. Furthermore, we developed a system to visualize the evaluation results, which is human-friendly and convenient to clinical application.

Key words Automatic Evaluation, EEG Recording, Artificial Intelligence, Electroencephalographer

1 INTRODUCTION

Electroencephalogram is the name of an important investigation carried out on most patients suspected of having nervous diseases. It is generated by nerve cells and recorded as continuous electrical activity. Such electrical activity changes according to the functional status of the brain. In the neurological medicine field, the feature of EEG recording is important for diagnosis, since it reflects the working state of human brain. The electroencephalographer interprets the EEG records by analyzing its time and spatial features with visual inspection. This work requires long-term EEG recording as well as sufficient experience of the consultant, because various conditions of the subject exist during the recording procedure.

A computer-assisted automatic interpretation system has been developed in our previous research, which makes it simplified for the EEGer to acquire the information of the subject. However, the accuracy of automatic interpretation is not as high as to be satisfied. This is mainly arisen from the uncertainty of EEG recording. Therefore, we are motivated to propose a method to estimate the state of the subject as well as evaluate the quality of EEG record. At the

Figure 1: Function of automatic evaluation for EEG recording

same time, we develop a system to visualize the evaluation results in order to make it conveniently used by the EEGer and the technician.

2 METHOD

2.1 Subject and data acquisition

The EEG data investigated in this paper is a part of consecutive EEG of a male, 69 years old, recorded in Kyoto University Hospital. The subject was placed in a supine position in a quiet, dimly lit room during recording. All electrodes were placed on the scalp according to international 10-20 system, and were referenced to the ipsilateral ear electrode. The 16-channel EEGs were recorded by an electroencephalography and stored in an optical disk. For analyzing, the digital data of EEG time series at the sampling interval of 5 msec were transformed into the Fourier components for each resolving frequency of 0.1 Hz. Features of each 5 sec EEG segment for each individual channel were thereby expressed by the periodogram

Table 1: Artifacts and relevant detection method

Artifact	Judgment conditions
Blink	**Existence of δ component** (1) $6\sqrt{S_\delta(F_{P1})} \geq 25[\mu V]$ (2) $6\sqrt{S_\delta(F_{P2})} \geq 25[\mu V]$ **Symmetry of the waveform** (3) $6\sqrt{S_\delta(F_{P1}+F_{P2})} \geq 50[\mu V]$ (4) $\dfrac{6\sqrt{S_\delta(F_{P1}-F_{P2})}}{6\sqrt{S_\delta(F_{P1}+F_{P2})}} \times 100 \leq 55[\%]$ **Extension to central region** (5) $\dfrac{6\sqrt{S_\delta(F_3)}}{6\sqrt{S_\delta(F_{P1})}} \times 100 \leq 85[\%]$ (6) $\dfrac{6\sqrt{S_\delta(F_4)}}{6\sqrt{S_\delta(F_{P2})}} \times 100 \leq 85[\%]$ (7) $\dfrac{6\sqrt{S_\delta(C_3)}}{6\sqrt{S_\delta(F_{P1})}} \times 100 \leq 78[\%]$ (8) $\dfrac{6\sqrt{S_\delta(C_4)}}{6\sqrt{S_\delta(F_{P2})}} \times 100 \leq 78[\%]$
Horizontal eye movement	**Existence of δ component** (1) $6\sqrt{S_\delta(F_7)} \geq 25[\mu V]$ (2) $6\sqrt{S_\delta(F_8)} \geq 25[\mu V]$ **Symmetry of the waveform** (3) $6\sqrt{S_\delta(F_7-F_8)} \geq 55[\mu V]$ (4) $\dfrac{6\sqrt{S_\delta(F_7+F_8)}}{6\sqrt{S_\delta(F_7-F_8)}} \times 100 \leq 91[\%]$ **Extension to central region** (5) $\dfrac{6\sqrt{S_\delta(T_3)}}{6\sqrt{S_\delta(F_7)}} \times 100 \leq 94[\%]$ (6) $\dfrac{6\sqrt{S_\delta(T_4)}}{6\sqrt{S_\delta(F_8)}} \times 100 \leq 94[\%]$
EMG	$6\sqrt{S_{EMG}(X)} \geq 10[\mu V]$, $\quad S_{EMG} : 35-50Hz$ $6\sqrt{S_{EMG}(X)} \geq 6\sqrt{S_\beta(X)}$, $\quad S_\beta : 13-25Hz$
Electrode	$6\sqrt{S_\delta(X)} \geq 25[\mu V]$ (X : respective electrodes on one side) $\sum \dfrac{6\sqrt{S_\delta(X-Y)}}{6\sqrt{S_\delta(X+Y)}} \times 100/11 \leq 30[\%]$ (X,Y : respective neighborhood electrode on one side of the scalp) $\sum \dfrac{6\sqrt{S_\delta(X-Y)}}{6\sqrt{S_\delta(X+Y)}} \times 100/8 \geq 50[\%]$ (X,Y : a pair of homologous electrodes on both sides)

parameters.

2.2 Function of the automatic evaluation system

The system we designed contains several functions according to clinical application. Basically, the automatic evaluation system for EEG recording consists of four sections: data acquisition, artifacts detection, wave evaluation and information display. Fig.1 represents the procedure of EEG recording and evaluating: a number of electrodes are pasted on the scalp of the subject and connected by the wires to electroencephalography; this machine records the electrical signals from the brain in order to produce permanent record; these record then is processed and evaluated by automatic evaluation system for EEG recording; if the record is judged as "qualified", the subject will keep the state for recording; otherwise, the technician will tell the subject to adjust the state. The objective is to realize the real-time processing. At present, we have approached to the on-line system development, which comprises detection of several artifacts, including eye blink, electromyogram, horizontal eye movement as well as ear-lobe reference electrode artifact; and also the evaluation of dominant rhythm - calculation of α wave parameters.

2.3 Detection of artifacts contamination

Various artifact contaminations are encountered frequently in the EEG records, such as blink artifact, EMG artifact, etc. These movements of eyes or muscles can cause large signals which may obscure the signals coming from the brain. They may also have effect on the EEG components. For example, the blink artifact contaminates the slow wave as well as the EMG artifact does the fast wave. The EEGer usually pays more attention to the artifact, and wish it could be avoided during recording. This is dependent on the cooperation of the subject. However, it is rather difficult for the subject to keep one state all over the

Table 2: Parameters explanation and relevant equations adopted for α wave evaluation

Parameters	$A_\alpha[\mu V]$	$A_\alpha(i) = \frac{1}{4}\sum_{j=1}^{4} 6\sqrt{S_\alpha(j,i)} \quad (j = O_1, O_2, P_3, P_4)$
	$D_\alpha[\%]$	$D_\alpha(i) = \frac{1}{4}\sum_{j=1}^{4} \frac{S_\alpha(j,i)}{S_T(j,i)} \times 100 \quad (j = O_1, O_2, P_3, P_4)$
	$F_\alpha[Hz]$	$F_\alpha(i) = \frac{1}{4}\sum_{j=1}^{4} f_\alpha(j,i) \quad (j = O_1, O_2, P_3, P_4)$
Normalized parameters	Φ_α^A	$\Phi_\alpha^A(i) = (A_\alpha(i) - \min A_\alpha(i))/(\max A_\alpha(i) - \min A_\alpha(i))$
	Φ_α^D	$\Phi_\alpha^D(i) = (D_\alpha(i) - \min D_\alpha(i))/(\max D_\alpha(i) - \min D_\alpha(i))$
	Φ_α^F	$\Phi_\alpha^F(i) = (F_\alpha(i) - \min F_\alpha(i))/(\max F_\alpha(i) - \min F_\alpha(i))$
Estimate value	B_α	$B_\alpha(i) = \Phi_\alpha^A(i)\Phi_\alpha^D(i)\Phi_\alpha^F(i)$
Relative ratio	R_α^B	$R_\alpha^B(i) = (B_\alpha(i))/(B_\alpha(std.)) \times 100\%$

recording. If the technician can notice the changeable condition and remind the subject instantly, it is very effective to reduce the artifacts contamination. Based on considering the respective features and results of visual artifacts inspection done by the EEGer, we have proposed the method to detect several artifacts successfully, including eye blink, electromyogram (EMG), horizontal eye movement (HEM), and ear-lobe reference electrode artifact. The judgment conditions for each artifacts are expressed with the equations shown in Table 1 [1].

In addition, the portion of artifacts is also valuable to the technician, such the position of EMG and that of the ear-lobe reference electrode artifact. Our method makes it possible to provide such information meanwhile. The name of the electrodes and the site of the head will be displayed when the corresponding artifact is detected.

2.4 Evaluation of dominant rhythm

The dominant rhythm is the most important item in EEG interpretation, and usually appears in the range of 8-13 Hz, which belongs to the α wave. It is considered as dominant EEG activity that consisted of waves with approximately constant period and with the maximum amplitude at the occipital or parieto-occipital regions of the head [2]. All parameters and relevant equations adopted for α wave evaluation are explained in Table 2, where the relative ratio is used to express the amount of existence of α wave. The value is given in percentage so as to contrast simply.

The appearance of α wave arouses the EEGer's attention greatly, because it is meaningful in neurophysiological diagnosis. In normal awake background EEG recording, α wave exists obviously. However, such dominant rhythm activity reduces by degrees due to the abnormality of the subject. For example, the α wave is hardly seen in markedly abnormal EEG record-

ing. Even in normal case, the state of the subject affects the α rhythm, such as decreasing for low vigilance level. Consequently, our proposed method to evaluate the α rhythm is significant to help the technician to distinguish the subject. Furthermore, it is available to caution the subject to keep high vigilance level. During the recording procedure, sometimes the subject may be in low vigilance level or in drowsy state, which results in the disappearance of dominant rhythm. In such case, the technician will tell the subject to open his eyes for a short while and then close his eyes again. After the subject recovers to waking state with closed eyes, the dominant rhythm will appear. And the evaluation of α wave is carried out continuously.

3 RESULTS

Currently, the automatic evaluation of EEG recording is realized through integrating artifacts detection and wave evaluation mentioned in the method. Fig.2 displays the EEG time series of the segment and evaluation results for each relevant one. The time series illustrates the continuous EEG record, from the 1st segment to the 4th, 5-second-long for each. The result of automatic evaluation for these segments is given in the lower table in the figure. Among the information, artifacts contamination and dominant EEG activity are involved. Blink artifact appears in most of the segments except the 4th; no other artifacts are detected. And the dominant rhythm exists in all of the segments.

4 DISCUSSION

The parameters adopted to calculate α wave are the general ones obtained by testing large amount of subjects. Similarly, the method of artifact detection is

Segment 1   Segment 2   Segment 3   Segment 4

Fp1-A1, F3-A1, C3-A1, P3-A1, O1-A1, Fp2-A2, F4-A2, C4-A2, P4-A2, O2-A2, F7-A1, T3-A1, T5-A1, F8-A2, T4-A2, T6-A2

$\_\_\_$|100 μ V
1 s

	Segment 1	Segment 2	Segment 3	Segment 4
Blink	Yes	Yes	Yes	No
Horizontal eye movement	No	No	No	No
EMG	No	No	No	No
Electrode artifact	No	No	No	No
Relative ratio for α wave [%]	97.4	78.5	61.0	60.4
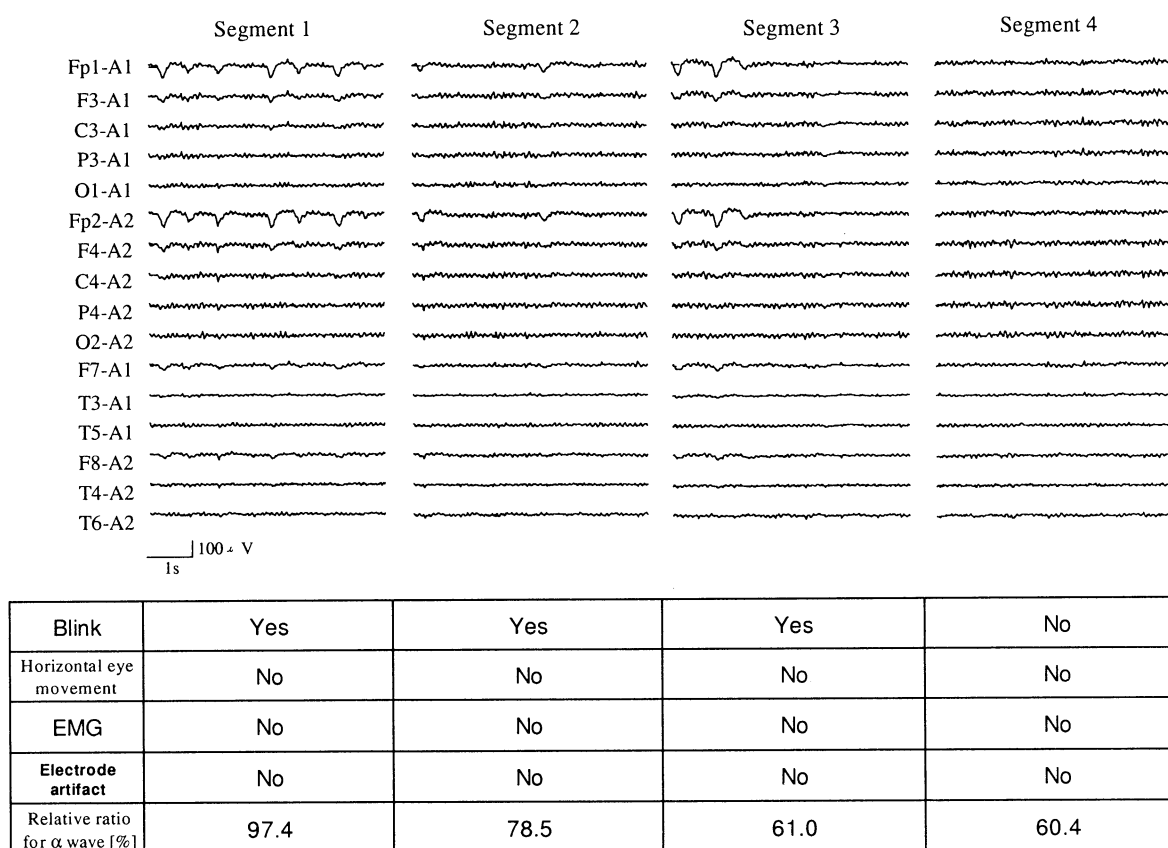
Figure 2: Results of automatic evaluation of EEG recording.

derived by considering the respective features and results of visual artifacts inspection done by the EEGer. So they can be regarded as the criteria to evaluate the EEG record.

In our research, we not only propose the method to evaluate the EEG recording, but also develop the system which visualizing all the evaluation results. With the information display system, it is very convenient and human-friendly to clinical application.

At present, the evaluation consists of artifacts detection and dominant rhythm investigation. Our further work is to realize such on-line system into real-time processing, and consider more items to improve the accuracy of identification.

5 CONCLUSION

We have proposed the method to estimate the state of the subject and have developed the system to evaluate the quality of EEG recording. In order to help the technician to recognize the subject's condition in-stantly throughout the recording, the on-line system will be evolved to real-time processing. Thus the quality of EEG recording is ensured, and the selection of appropriate segments for integrative EEG interpretation is thereby simplified. In addition, the system is developed based on artificial intelligence of electroencephalographers and human-friendly to the users.

References

[1] T. Sugi, M. Nakamura, A. Ikeda, et al, "Automatic detection of artifacts for pre-processing of automatic EEG interpretation" (in Japanese). Medical Electronics and Biomedical Engineering, vol. 33-3, pp.203-213, 1995.

[2] M. Nakamura, H. Shibasaki, K. Imajoh, et al, "Automatic EEG interpretation: a new computer-assisted system for the automatic integrative interpretation of awake background EEG". Electroenceph. clin. Neurophysiol., vol.82, pp.423-431, 1992.

Mutual Conversion of Sensory Data and Texts by an Intelligent System IMAGES-M

Daisuke HIRONAKA   Seio ODA   Kouichi RYU   Masao YOKOTA

hironaka@fjct.fit.ac.jp   oda@fjct.fit.ac.jp   ryu@fit.ac.jp   yokota@fit.ac.jp

Fukuoka Institute of Technology

3-30-1 Wajiro-higasi,Higasi-ku,Fukuoka-shi,Japan

Abstract

An intelligent system IMAGES-M, especially one of its functions, mutual translation of sensory data and natural language texts is briefly sketched. IMAGES-M is apparently one kind of expert system performing cross-media translation via media-free conceptual descriptions based on the Mental Image Directed Semantic Theory (MIDST).

Key words :cognitive robotics, symbol grounding problem, sensory data, natural language, cross-media translation

1 Introduction

Development of intelligent robots and rapid increase of aged societies have brought serious necessity of such systems that should facilitate mutual translation of sensory data and linguistic expressions. They are expected to help people, esspecially with some defected sense-organ, by translating sensory data into words such as "Pungent smell is sensed in the refrigerator!" and otherwise to enable people to order a robot to work by words such as "Search the room for a vari-colored object."

Such a system should be equipped with a certain semantic understanding process where words are associated with external information, namely, sensory data. Most of the meaning representation schemes adopted in the traditional natural language understanding systems would be inadequate. Because they are designed exclusively to describe semantic relations among verbs and nouns, that is, deep case relations are still too linguistic or macroscopic to represent fine spatiotemporal relations brought by external information [1, 2].

As another approach, Yokota,M. et al have proposed the Mental Image Directed Semantic Theory (MIDST) [3] which treats word concepts in association with non-linguistic mental images of the external or physical world and formalizes them on the basis of the first order predicate logic. Here is briefly sketched an intelligent system based on MIDST, so called IMAGES-M, especially one of its functions, mutual translation of sensory data and natural language texts [4]. This project consists of three components as follows:

(C-1) Conceptual analysis and description of words based on a mental image model.

(C-2) Grounding words on real sensory data guided by their conceptual descriptions.

(C-3) Mutual conversion between real sensory data and texts via conceptual descriptions.

2 Mental Image Model

The Mental Image Directed Semantic Theory treats word concepts in association with mental images modeled as "Loci in Attribute spaces" [3]. An attribute space corresponds with a certain measuring instrument just like a barometer, a map measurer or so and the loci represent the movements of its indicator.

A general locus is to be articulated by "Atomic locus" depicted in Fig.1 and formalized as Expression (1) in first-order predicate logic, where "L" is a predicate constant.
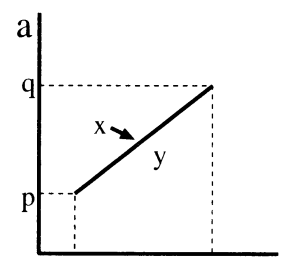
Fig.1 Atomic locus

$$L(x, y, p, q, a, g, k) \qquad (1)$$
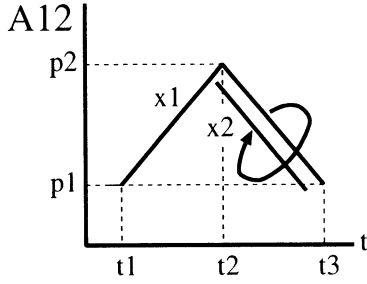
Fig.2 Locus of the English word "fetch"

Expression (1) is called "Atomic locus formula" and interpreted as follows 治而
"Object 'x' causes Attribute 'a' of Object 'y' to keep $(p = q)$ or change $(p \neq q)$ its values temporally $(g = Gt)$ or spatially $(g = Gs)$, where the values 'p' and 'q' are relative to Standard 'k'." When $g = Gt$, the locus indicates monotonous change or constancy of the attribute in time domain and when $g = Gs$, that in space domain. The former is called a temporal event and the latter, a spatial event. For example, the motion of the 'bus' represented by S1 is a temporal event and the ranging of the 'road' by S2 is a spatial event whose meanings or concepts are formalized as Expression (2) and (3), respectively, where the attribute is "physical location (A12)".

$(S1)$ *The bus runs from Tokyo to Osaka.*

$$(\exists x, y, k)L(x, y, Tokyo, Osaka, A12, Gt, k)$$
$$\wedge bus(y) \qquad (2)$$

$(S2)$ *The road runs from Tokyo to Osaka.*

$$(\exists x, y, k)L(x, y, Tokyo, Osaka, A12, Gs, k)$$
$$\wedge road(y) \qquad (3)$$

Expression (4) is the conceptual description of the English word "fetch", depicted in Fig.2, implying such a temporal event that 'x1' goes for 'x2' and then comes back with it, where 'Π' and '·' are instances of the tempo-logical connectives, standing for "Simultaneous AND" and "Consecutive AND", respectively. In general, a series of atomic locus formulas and tempological connectives is called 'Locus formula'.

$$(\exists x1, x2, p1, p2, k)L(x1, x1, p1, p2, A12, Gt, k) \cdot$$
$$(L(x1, x1, p2, p1, A12, Gt, k) \sqcap$$
$$L(x1, x2, p2, p1, A12, Gt, k))$$
$$\wedge x1 \neq x2 \wedge p1 \neq p2 \qquad (4)$$

3   IMAGES-M

The intelligent system IMAGES-M, still under construction, is intended to facilitate integrated multimedia information understanding, including multisensory data fusion so called in robotics. Figure 3 shows the configuration of IMAGES-M, consisting of such seven major modules as follows:

(1)   Text Processing Unit (TPU),

(2)   Speech Processing Unit (SPU),

(3)   Picture Processing Unit (PPU),

(4)   Animation Processing Unit (APU),

(5)   Sensory Data Processing Unit (SDPU),

(6)   Inference Engine (IE), and

(7)   Knowledge Base (KB).

The five processing units (i.e. TPU, SPU, PPU, APU, and SDPU) mutually convert information media (i.e. texts, speeches, still pictures, animations and sensory data) and conceptual descriptions in aid of IE and KB. IMAGES-M is apparently one kind of expert system performing cross-media translation via media-free conceptual descriptions, namely, locus formulas, and miscellaneous combinations of its processing units bring forth various types of cross-media translation in aid of IE and KB. For example, the function of mutual conversion of sensory data and texts is the collaboration of SDPU and TPU.

4   Mutual conversion of sensory data and texts

IMAGES-M is intended to take in data from various kinds of sensor and to translate them into locus formulas and in turn into texts. Of course, in advance to such performances, the system must learn word concepts grounded on real sensory events. Although the methods presented here are explained exclusively for the cases of vision, they are common to the cases of the other kinds of sensation, i.e. touch, taste, etc.

4.1   Word grounding on real sensory data

This function is performed by SDPD, TPU, IE, and KB. IMAGES-M acquires word concepts by grounding words on real sensory data via locus formulas interactively with a human instructor. For a very simple example, acquisition of the English verb concept 'redden' is briefly described as follows:
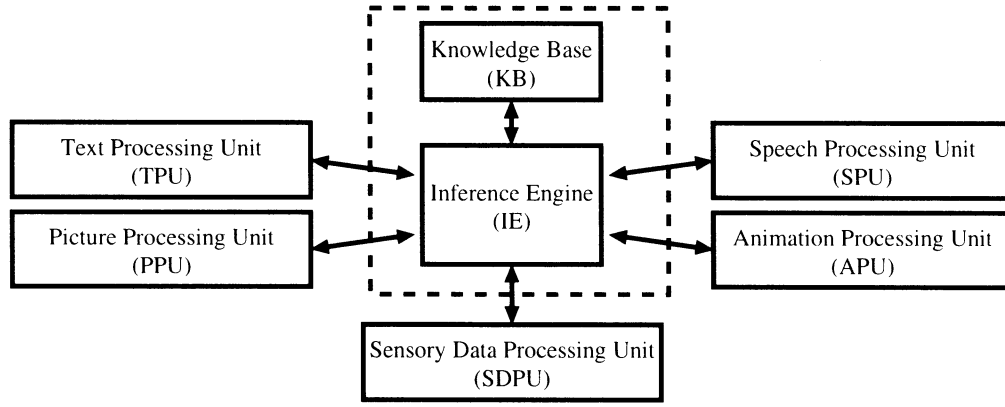
Fig.3 Configuration of IMAGES-M

(STEP-1) The human instructor puts into the system, through TPU, the sentence pattern " x redden(s)" and its concept pattern given by Expression (5).

$$L(y, x, v1, v2, a, g, k) \qquad (5)$$

(STEP-2) The human instructor informs the system of such a set of facts as follows:
$\{y = \ '\varphi'(=\text{Don't care.}),$
$a = \ 'A32' \ (=\text{Care about color.}),$
$g = \ 'Gt' \ (=\text{Care about temporal change.}),$
$k = \ 'Mansel' \ (=\text{Use Mansel color solid.})\}$

(STEP-3) Through a video camera connected to SDPU, the human instructor shows the system a pair of pictures {P1, P2}, ordered in time, where in P1 an object O1 with its color C1, not red, and in P2 the same object O1 with its color red.

(STEP-4) The human instructor shows another pair of pictures {P3, P4}, where in P3 another object O2 with its color C2, neither C1 nor red, and in P4 the object O2 with its color red.

(STEP-5) The system generalizes/conceptualizes the input data as Expression (6) for "x redden(s)", which is a special case of inductive inference by IE.

$$(\exists x, v1, v2)L(\phi, x, v1, v2, A32, Gt, Mansel)$$
$$\wedge v1 \neq v2 \wedge v2 = Red \qquad (6)$$

4.2 Text generation from sensory data

This function is realized as the processing flow from SDPU to TPU in aid of IE and KB. Once IMAGES-M acquires the concept of 'redden' in such a way above, it

can employ all its knowledge and generate a set of sentences corresponding with another pair of input pictures {P5, P6}. For example, assume a yellow disk in P5 and a red disk in P6, respectively, then the following is the output sentences of IMAGES-M:

The color of the disk changes.
The disk changes its color.
The color of the disk changes from yellow to red.
The disk turns red.
The yellow disk turns red.
The disk reddens.
The yellow disk reddens.

.................,

where the concepts of 'change', 'turn', 'disk', etc. were in advance acquired by IMAGES-M. These sentences can be generated by unifying Expression (7) below and the verb concepts memorized in KB, which is a conventional case of deductive inference by IE.

$$L(\phi, x, Yellow, Red, A32, Gt, Mansel) \wedge disk(x) \qquad (7)$$

4.3 Sensory data generation from texts

The processing flow from TPU to SDPU should be this function. However, it is actually very difficult to synthesize all kinds of real sensation by computers, which leads to employing virtual reality technologies or the other alternatives. IMAGES-M utilizes PPU and APU instead of SDPU and makes them translate locus formulas into pseudo-visions, i.e. still pictures and animations. Figure 4 shows an example. The collaboration of TPU and PPU/APU is digested as follows:

(STEP-1) TPU takes in a text and translates it into a conceptual description, so called, 'Text Meaning

The lamp above the chair is small.
The red pot is 1m to the left of the chair.
The blue big box is 3m to the right of the chair.

(a) Input text
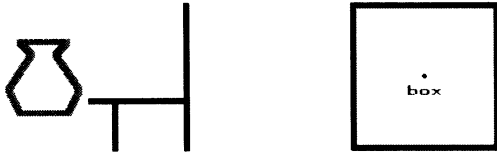
(b) Output picture

Fig.4 Example of text-picture translation
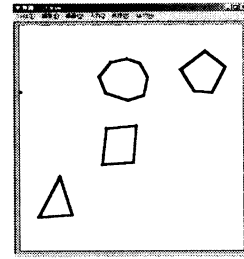
(a) Input picture

The octagon is to the upper right of the triangle.
The octagon is above the quadrangle.
The triangle is to the lower left of the octagon.

(b) Output text

Fig.5 Example of picture-text translation

Representation (TMR)', where syntax and meaning dictionaries in KB are utilized through IE.

(STEP-2) IE, using KB, deduces from the TMR a conceptual description adaptive for picture/animation, so called, 'Picture Meaning Representation (PMR)' which reflects all the functional constraints of the output device, color TV monitor. For a very trivial example, such a device can display 'a red candy', but not 'a sweet candy'. That is, a PMR does not contain any locus formula of taste, smell, or so but of vision.

(STEP-3) PPU/APU interprets the PMR as a picture / animation. The choice between PPU and APU depends on the verb concepts contained in the text. For example, the PMRs for S1 and S2 are sent to APU and to PPU, respectively.

The reverse process, namely, picture-to-text translation has been also realized, whose example is shown in Fig.5.

5 Conclusion

The issues of our project by now are as follows:

(I1) About 40 attributes (weight, temperature, shape, etc.) were found out.

(I2) A greater part of the words of the attributes 'color', 'lightness' and 'temperature' were successfully grounded on real sensory data.

(I3) The mutual conversion between texts (in Japanese, Chinese or English) and pictorial patterns were almost successfully implemented.

The very aim of our research is to solve the symbol grounding problem [5] in cognitive robotics. Such simple word concepts as can be represented in a single atomic locus formula are easily grounded on real sensory events. However, for much more complicated ones such as 'fetch', the system must be equipped with further sophisticated pattern recognizers.

References

[1] Fillmore, C.(1968): "The case for case, in Universals in linguistic theory (eds. E. Bach and R. Harms)", Holt, Rinehart and Winston.

[2] Jackendoff, R.(1988): "Semantics and Cognition", The MIT Press.

[3] Yokota,M., et al(1991): "Mental-image directed semantic theory and its application to natural language understanding systems", Proc. of NL-PRS'91, pp.280-287.

[4] Oda,S. et al(2001): "Conceptual Analysis and Description of Words for Color and Lightness for Grounding them on Sensory Data", Trans. of JSAI, 16-5-E, pp436-444.

[5] Harnad, P.(1990): "The Symbol Grounding Problem", Physica D,42, pp. 335-346.

Automatic Determination of Sleep Stages by the Multi-valued Decision Making Method with Knowledge Enhancement

Masatoshi NAKAMURA*, Bei WANG*, Takenao SUGI† and Fusae KAWANA‡

* Department of Advanced Systems Control Engineering, Saga University,

Honjomachi, Saga 840-8502, Japan

†Department of Electrical and Electronic Engineering, Saga University,

Honjomachi, Saga 840-8502, Japan

‡Department of Clinial Physiology, Toranomon Hospital Tokyo, Japan

E-mail: wangbei@cntl.ee.saga-u.ac.jp

Abstract

This study presents an automatic determination system to computerize sleep stage scoring. The main method of multi-valued decision making, which was proposed by some of the authors in the recent years, was based on the Bayes rule and statistical knowledge. However, like the foundamental problem in most stochastic progressing is learning from the experience, to obtain enough knowledge takes an important role for the whole system. Enhancements for database construction were taken in this study to improve the accruacy of automatic sleep stage determination. Comparing with the scoring criteria published by Rechtschaffen and Kales, a classification of open and close eyes during awake was adopted to meet a well fit between features and stages. In addition, the distributions of features for stage discriminatation were made through Cauchy distribution instead of Gauss distribution to avoid the calculation errors caused by the digital computers. The overnight sleep recordings of several subjects were tested in this study. The result was quite satisfied comparing with the stage scoring investigated by EEGers (clinician) in Toranomon Hospital.

1   Introduction

Sleep is essential for the normal, healthy functioning of the human body. Although sleep may seem like a steady state, it actually consists of several stages that cycle thoughout the night from light sleep to deep sleep. Sleep stage scoring is an important diagnosis way for the patients suffered by sleep disorders. But to make visual investigation for one's overnight sleep is a rather laborious task, and today computerized sleep stage scoring system becomes helpful and necessary for sleep studies.

Indeed, many researches had been done in this field. The main technique they used was relied on the detection of characteristic waveform in different stages [4][5]. In this study, we considered the determination problem in a different way and try to solve it as a problem of decision making. In our daily life we may face with similar problems like decision making. Usually we made judgement based on the experience in our mind and the current situation. It is due to this idea that some of the authors proposed the multi-valued decision making method [2] in order to solve the decision making problem automatically. By using this method, the determination of current segment would be done based on the information from previous segments and the current characterisitcs.

However, the accuracy of our system was mostly depending on the knowledge obtained through learning processing. In this study, in order to obtain enough knowledge, a classification of open and close eyes during awake was adopted to meet a well fit between features and stages. In addition, the distributions of features for stage discriminatation were made through Cauchy distribution instead of Gauss distribution in order to avoid the calculation errors caused by the digital computers.

2   Method

2.1   Sleep stages

The guideline published by Rechtschaffen and Kales in 1968 [1] was used frequently as standardization of terminology and sleep staging methods in humans. The types of brain waves (based on amplitudes and

frequencies) determine the stages of sleep. Generaly, sleep is "staged" into 20- or 30-second epochs, and these epochs are considered both independently and in the context of preceding and succeeding epochs. Usually sleep were discriminated into stage awake, REM and NonREM, where NonREM was constited with stage 1, 2, 3 and 4. But there was a major problem in awake that EEG showed predominant alpha activity (8-13Hz) as the eyes remain closed while significantly reduced as the eyes opened [3]. In this study, we classified stage awake into open and close eyes in order to make a well fit between features and stages.

2.2 Stage determination

In sleep stage scoring, the continuous recording of one's overnight sleep recording would be seperated into 30- or 20-second and the judgement would be done for each segment. The activity of stage scoring was described by Fig.1. Skillful clinicians had enough experience and knowledge in sleep studies. As sleep was a continuous state, stage would be determined not only from the current characteristic but also the information from previous segment. When the EEGer investigated the sleep recordings, he or she would analyze on the current characteristics and preivous information to make judgement. Finally, the most matched stage would be the result of stage scoring. In this section, the method of multi-valued decision making would be adopted in order to automize the laborious work in stage scoring by using the theory and knowledge in statistics and probability.

2.2.1 Database construction

The polysomnograph (PSG) measurement which includes electroencephalagram (EEG), electromyogram (EMG), electrooculogram (EOG) and other signal types was used in sleep study. We extracted characteristic parameters for stage discrimination from the periodogram of each signal channle based on the guidelines of Rechtschaffen and Kales, they are:
Duration: the ratio of power in the frequecny band($\delta 1$:0.5-2Hz, α :8-13Hz) to the EEG band(T :0.5-25Hz);
Amplitude: the square root of power in the prequency band θ :2-7Hz, high frequency:25-50Hz in EEG channle;
Amount: the power of the frequency band (2-10Hz) in LOC, ROC and LOC-ROC that represents the eye-movements and frequency band (25-100Hz) in chin-EMG that represents chin tone.
The purpose of database construction is to obtain

Experiences: characteristics of each stage
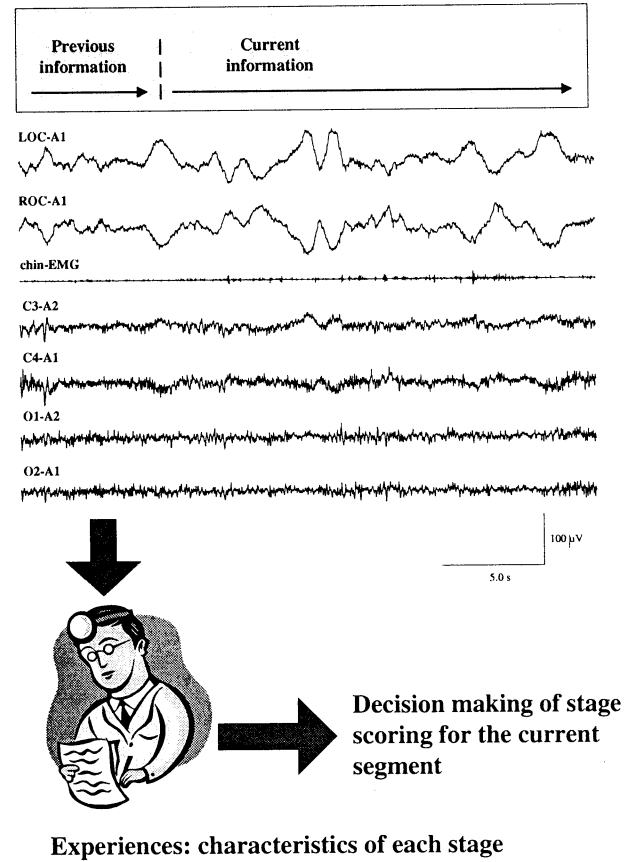
Figure 1: EEGer scored stage of the current segment based on his or her medical experience. The current characteristics would be investigated as well as the previous information would be refered when necessary.

the statistical distribution between features and sleep stages. The sleep recording which are investigated by EEGer (clinician) by visual analyse were used as learning data. Stage awake was investigated or classified into open eyes awake and close eyes awake, due to the amount of alpha components (8-13Hz).

The distributions of parameter for stage discrimination were approximated by Cauchy distribution. The reason why we did not use Gauss distribution was due to the calculation limitation of the digital computers. We have found the digital computers had calculation limitation when some value was located on the tail of the distribution. Cauchy distribution has infinite variance, the tails were much heavier than Gauss distribution to the digital computers. So that, for stage

ζ^i, the parameter distribution would be

$$f(y) = \frac{b}{\pi((x-m)^2 + b^2)} \qquad (1)$$

where m was the location parameter, b was the scale parameter of Cauchy distribution. They were undefined and obtained by approximating to the histogram.

2.2.2 Automatic determination

During automatic sleep stage determination, the method of multi-valued decision making would be adopted in order to realize automatic determination. At segment k, we obtained the conditional probability of stage ζ^i by

$$P_{k|k}(\zeta^i) = \frac{f(y_k|\zeta^i)P_{k|k-1}(\zeta^i)}{\sum_{j=1}^{7} f(y_k|\zeta^j)P_{k|k-1}(\zeta^j)} \qquad (2)$$

where $f(y_k|\zeta^i)$ was the combined probability of features. Then the principle of decision making was defined in Eq. 3 by choosing the maximum value of conditional probability,

$$\zeta^* : max(P_{k|k}(\zeta^i)). \qquad (3)$$

The predicted probability $P_{k+1|k}(\zeta^i)$ for next segment k+1 would be calculated based on transitinal probabilities t^{ij} from stage i to stage j and the conditional probability of k-th segment, as

$$P_{k+1|k}(\zeta^i) = \sum_{j=1}^{7} M_{ij}P_{k|k}(\zeta^j) \qquad (4)$$

With an initial predicted probability $P_{1|0} = 1/n$, $i.e.$ each stage has the same value of predicted probability, and a given transitional probability matrix t^{ij}, $i.e.$ the probability of stage change during one's sleep prcessing. The whole algorithm would be iterated through the calculation of the conditional probability $P_{k|k}$ and the predicted probabilty $P_{k+1|k}$. Decision making of ζ^* for segment k would based on the conditional probability $P_{k|k}(\zeta^i)$.

3 Result

3.1 Database construction

The subjects we used were the overnight sleep recordings recorded in the Toranomon Hospital. The training data during database construction was one of
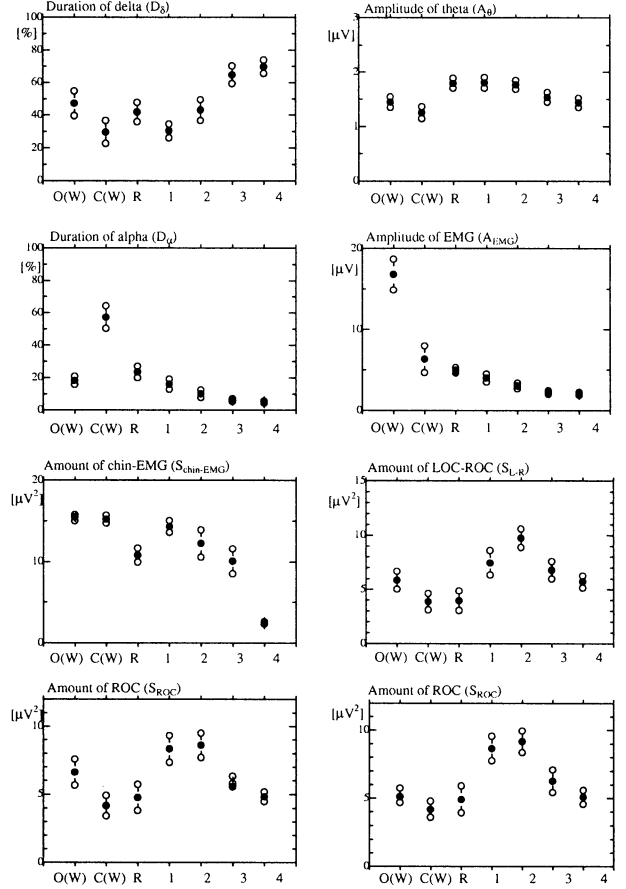
Figure 2: The parameter distributions for each stage were given respectively, where the black dots refered to position parameter m, the empty dots refered to scale parameter b.

the subjects (54 years old, male) with 8-hour sleep recording. Based on the visual scoring made by EEGer, the distributions of all the parameter in each stage were showed in Fig.2. The alpha components in open and close eyes during awake were quite seperated from each other, with the Cauchy distribution $(m \pm b)$ of 19.3 ± 2.35 and 58.7 ± 8.57 respectivly.

3.2 Automatic determination

The test sujects included the training data (subject 1) and another two subjects (subject 2 and 3), all were male with age of 54, 60 and 50 respectively. The results of automatic sleep stage determination were evaluated in Table 1. The recognition of stage awake among the three subjects was quite satisfied, where stage 1/2 refered to light sleep and stage 3/4 refered

Table 1: Evaluations of sleep stage determination

	Subject 1	Subject 2	Subject 3
Awake	90.9%	92.2%	90.4%
REM	88.0%	64.4%	52.5%
Stage 1/2	87.0%	68.4%	79.0%
Stage 3/4	89.2%	95.0%	99.3%

to deep sleep.

4 Discussion

4.1 Multi-valued decision making

The method used in this study was unlike the way of many other researchers had done in automatic sleep stage scoring. The main technique they used was relied on the detection of characteristic waveform in sleep stages [4][5]. Althrough characteristic wavefomrs detection maybe one of a useful way for stage discrimination, but a well analyse on the stages was also necessary and important for their method as well as to find appropriate thresholds for each waveform through a large amount of tests. In our method, the learning processing and automatic determination were not depend on the thresholds but through the distributions of effective parameters which included all the actual informations from subjects. It is rather like an expert system when applied to different subjects.

4.2 Classification of open and close eyes during awake

In 1968, Rechtschaffen and Kales published guidelines[1] for the standardization of terminology and sleep staging methods in humans. But today it is not effective enough for sleep studies. The insuffient number of stages maybe one of the problem. As the parameters were supposed to be assembled around a certain value, it would be difficult to describe stage awake with alpha component due to the obvious difference during open eyes state and close eyes state. In order to make a well fit between features and stages, we classified stage awake into the two sub-states.

4.3 Cauchy distribution and calculation limitation

The probability density functions between features and stages took important roles in the calculation of conditional probability. As we have known the variance of Gauss distribution was limited in a certain area around the mean value, the tails would be too small to the digital computers. In the other hand, it is inevitable that some unknown value would located at the tails of the distributions during automatic determination algorithm. In order to avoid the calculation errors, we use Cauchy distribution instead of Gauss distribution which had an infinite variance and the shape was similar to Gauss distribution.

4.4 Future work

Further research should be done to develop a more accurate automatic determination system. Usually, the EEG or other signals would be contaminated with artifacts. Some of the aritfacts would easy to make mistake during the determination processing, when they had similar property in a certain frequency band such as electrode artifacts accompanied with slow waves, or when the amplitude was too huge that important information can not be extracted effectively such as body movements. Sometime, even the EEGer would feel difficult when faced with contaminated signals. So that processing should be taken into account not only to reduce the amount of mis-judge stages but also to help EEGer to analyze on the real EEG through artifact eliminating.

References

[1] A. Rechtschaffen and A. Kales, *A Manual of Sandardized Terminology, Techniques and Scoring System for sleep stages of Human Subjects California, Brain Information Service*, 1968.

[2] Masatoshi Nakamura and Takenao Sugi, "Multi-Valued Decision Making for Transitional Stochastic Event: Determination of Sleep Stages through EEG Record" *Transactions on Control, Automation and Systems Engineering*, Vol. 3, No. 2, June, 2001.

[3] Ou Bai, Masatoshi Nakamura, Akio Ikeda and Hiroshi Shibasaki, "Automatic detection of open and closed eye states in the electroencephalographic (EEG) record for background EEG interpretation by the trigger method" *Frontiers Med. Biol. Engng*, Vol. 10, No. 1, pp. 1-15, 2000.

[4] J.R. Smith, I. Karakan and M. Yang, "Automatd analysis of the huamn sleep EEG" *waking and Sleeping*, vol. 2, pp. 75-68, 1978.

[5] Rajeev Agarwal and Jean Gotman "Computer-Assisted Sleep Staging" *IEEE Trans. Biomed. Eng.*, Vol. 48, pp. 1412-1423, 2001.

Adaptive Communication among Collaborative Agents: Preliminary Results with Symbol Grounding

Yoosook Lee[†]   Jason Riggle[§]   Travis C. Collier[†]   Edward Stabler[§]

Charles E. Taylor[†*]

Abstract

Communication among adaptive agents can be framed as language acquisition and broken down into three problems: *symbol grounding, language learning, and language evolution*. We propose that this view clarifies many of the difficulties framing issues of collaboration and self-organization. Additionally, we demonstrate simple classification systems that can provide the first step in grounding real-world data and provide general schema for constructing other such systems. The first system classifies auditory input from frog calls and is presented as a model of grounding objects. The second system uses the minimum description length framework to distinguish patterns of robot movement as a model of grounding actions.

1 Introduction

1.1 The Vision and Approach

Adaptive collaboration holds great promise for artificial life and engineering generally. At present there is no good theory of self-organization and collaboration, though the benefits are obvious. These include concurrent specialization by different individuals; distributed presence (for triangulation or "extending the eyes and ears of a small party"); robustness of individuals or functions that can be assumed by others; ability to avoid high risk or high cost behaviors until needed, among others.

The long term goal of our research is to produce a collection of heterogeneous robots that learn about their environment, communicate with one another about it, affect the environment in ways that take advantage of one another's special abilities and circumstances, and report to human observers.

The work described here is directed toward creation of robots, or more abstractly agents, which can learn to bind symbols to sensory patterns from

---

[*]taylor@biology.ucla.edu

[†]Department of Organismic Biology, Ecology, and Evolution, University of California Los Angeles

[§]Department of Linguistics, University of California Los Angeles

their environment and to evolve a language that will enable them to use those symbols for meaningful communication among themselves. This language will also provide a logic to facilitate reasoning.

1.2 Language at the Core

At this point the principal challenges are to further develop the cognitive and linguistic features of the system. It is desirable that agents be able to acquire their own language. First, an acquired language can evolve over time to adapt its structure to various types of noise and different types of information to be transmitted, thus providing a truly flexible communications channel. Also, agents with heterogeneous sensor modalities can map the same symbol to an environmental stimulus despite having very different internal representations. This requires agents to acquire their own symbol-to-meaning mappings, based on how their own observations correlate with the symbols that other agents transmit. Finally, language also provides a logical manipulation facility for cognitive reasoning. Our approach has been explained in more detail elsewhere[1].

1.3 Language Acquisition

Language acquisition provides a well explored framework for many of the problems in designing collaborative agents. From a linguistic perspective, the primary problems are: symbol grounding, language expression and learnability, and language evolution.

The problem of *symbol grounding* is, given a finite sequence of sensory impressions, each with potentially infinite detail and accompanied by a sentence, how does one come to associate particular lexical items intended meanings[2]? Symbol grounding, in the most general case, is known to be intractable. However, Siskind[3] and others have shown that when there is a consistent shared hierarchy of salience, then computational systems can overcome the grounding problem in quite general circumstances.

Language learning is the identification of a possibly infinite language and its syntactic rules from a finite sequence of examples. For a long time, reasonably expressive languages were considered unlearnable[4]; however, recent progress by Angluin[5], extended by Kanazawa[6], Denis[7], and Stabler[8, 9], has demonstrated that a class of quite expressively powerful languages can be learned.

The third problem, *language evolution*, involves determining how the language used by a collection of agents will change over time. This aspect is closely related to just what information needs to be expressed and to the kinds and quantities of noise the agents have to deal with. Parameters such as the maximum amount of communication channel noise that can exist before the agents lose their ability to communicate in a common language have been determined for simple regimes[10]. Doing such analysis for the system we envision may not be analytically tractable, but empirical studies informed by the theoretical results for simpler systems are possible.

In this paper we will focus our discussion on the the first of these language acquisition problems – symbol grounding.

1.4 Symbol Grounding

In symbolic systems, transfer of information is accomplished by associating symbols with words, gestures, or other behaviors that can be sensed by oth-

ers. Establishing this association between a symbol and the information it conveys is called *grounding*. In an adaptive system, the meaning of a symbol might vary, and so a language learner needs to somehow ground its linguistic experience in the cognitive domain, establishing the semantic values of the symbols.

Grounding, the learning of culturally common semantic values on the basis of similar experience, also allows for language change. Like other instances of phenotypic plasticity, this can allow a single organism to adapt and survive where a fixed system might fail. Humans can quite quickly adapt to widely different sorts of vocal tracts and other synthesizers in spite of vastly different acoustic properties, and even when vocal communication becomes impossible, there are often other means. Robots can have similar capabilities. Note in particular that in the situation where two robots already have had a common language and previous communication so that each has a model of the other, then if a change in language is needed, grounding the changed language can be very quick and efficient.

Various approaches to the roles of perception and language adaptation in grounding problems have been explored[11, 12, 13]. Here we focus on ideas in simple acoustic classification and the *minimal description length* (MDL) framework[14, 15, 16].

2 Grounding Objects

The first task any symbolic communication system must solve is how to classify non-symbolic sensory input. This is a prerequisite for further processing functions, such as localization. For the communication that allows collaboration to occur, a classification system is needed to bind actual perceptual information into symbolic tokens that can carry the semantic meaning.

Our model for this step has been an aural recognizer that quickly discriminates between different species of frog calls with reasonably low error. Frog calls provide a good model because as they are typically simple, short, and well deliminated. As such, they are useful in a variety of model situations, including localization[17]. Besides their intrinsic interest to biologists, frogs are often a bellwether of ecological disturbance such as pollution

and climate change, so that, monitoring species distributions and abundances could provide an early warning of environmental problems.
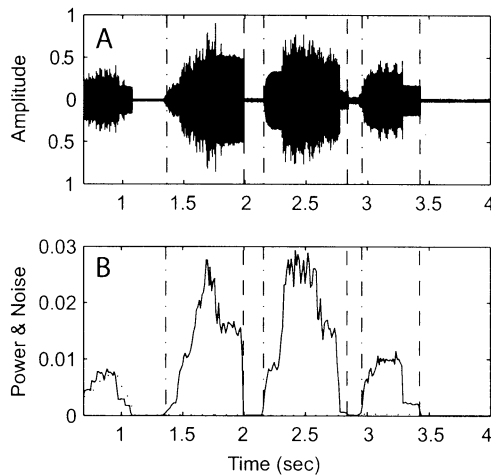
Figure 1: Signal detection of frog calls. Detected starting points (— · —·) and ending points (— — —). **A**. Amplitude of input source. **B**. Power and estimated noise (· · ·) derived from amplitude data.

Our system works as follows. Given raw auditory input the agent starts off by determining the beginning and ending points of potential *signals* as shown in Figure 1*A*. This is accomplished by keeping a running estimate of noise and looking for increases (or decreases) in the signal-to-noise ratio as depicted in Figure 1*B*. When a signal has been detected, the spectrogram for the signal is computed and compared to spectrograms of previously identified sounds. This signal spectrogram comparison is simply the maximum cross correlation coefficient which has been shown to properly correlate to the probability that two samples are identical except for additive white-noise[18]. If the new signal does not correlate well with any previously identified sounds, it is considered to be a new sound, and is saved so that additional signals can be compared to it as well as to the previously identified ones.

Figure 2 shows the first results from real data (the published recorded calls of frogs and toads in French Guyana[19]); 1507 potential signals from 62 different frog species. Using 0.7 as the similarity threshold for matching and 0.65 as the threshold
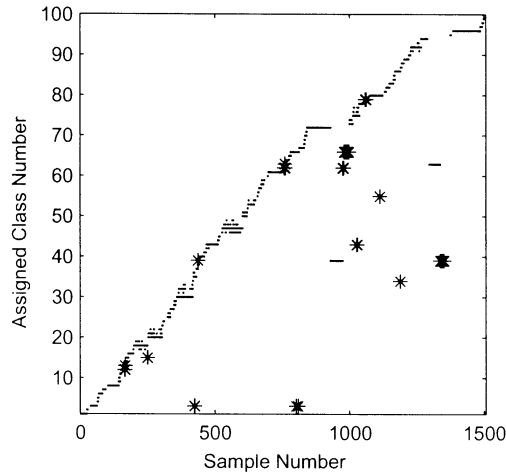
Figure 2: Classification performance from frog calls. Correct classification (·); Incorrect classification (∗). Each new signal is either assigned to an existing class or creates a new class.

below which the signal is considered new, we failed to classify 15% of the signals and misclassified less than 5%.

This general strategy of detecting potential signals, transforming the signal data to extract relevant features for comparison, and then applying a statistical comparison to assign the signal to a class of known signals (or else create a new class of known signals) is easily extended. More sophisticated techniques are possible, and we are currently collaborating with engineers and computer scientists to explore pitch, duration, power-spectra, and other measurements. Multiple metric analysis opens up the possibility of doing hierarchical classification, including subclasses that correspond to adjectives.

Although far from perfect, this simple method works much better than one would expect after reading the very discouraging literature on related tasks such as speech-recognition[20]. Interesting patterns that can potentially distinguish real biological hypothesis are discernible even at this stage, although clearly more work needs to be done. We are currently working with herpetologists to explore strength of species relatedness, environmental conditions, and inter-specific competition for bandwidth in determining the similarity between differ-

ent frog species calls.

3 Grounding Events

Sequences of symbolic data, either produced by object grounding or directly from sensors, can also be grounded with linguistic inputs to produce the equivalent of verbs and adverbs. Our work in this direction is focused on analyzing behavioral data from animals and robots using the MDL as a basis for generalization.

3.1 Minimal Description Length

The MDL framework[14, 15, 16] is essentially an implementation of standard Bayesian induction, where the priors are set by the representational complexities of the hypotheses. The basic idea is that the learner prefers the simplest description of the evidence, where the measure of simplicity is given by the representational complexity of the hypothesis together with its encoding of the evidence. A hypothesis that specifies a sufficiently general tendency in the data will make the encoding of the data smaller, for an overall improvement in simplicity.

Many different learning models can be represented in this framework, and they have proven useful already in theories of perception[21] and language learning[22, 23, 24, 25]. We have deployed these models in a few of preliminary domains: automata induction[26], and recently in identifying wall-following and other robotic behavior.

Finding an absolutely minimal description is impossible and unnecessary. The best compression of a set of finite strings is not, in general, a computable function[27]. Moreover, for simple, finite strings, we are often interested in differences smaller than the constant bound distinguishing different universal machines[28]. For our purposes, a simple and efficient metric of hypothesis complexity will be most useful in guiding generalization.

3.2 A Simple Example of MDL

A simple example is presented in Figure 3 which illustrates sensors distributed in a room. A robot wandering around the room activates the sensor in the region it occupies. The following stimulus sets
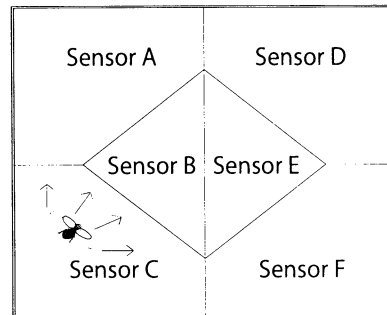
Figure 3: A robot wandering around a room where array of sensors are distributed.

include elements that represents the movement of a robot at each time, kindly provided to us by Brooks and Friedlander.

$$A = \{[\ a,\ a,\ c,\ f,\ f,\ f,\ d,\ d,\ f,\ c,\ a,\ a\ ],$$
$$[\ c,\ c,\ f,\ f,\ c,\ b,\ b,\ b,\ b,\ f,\ c,\ a\ ],$$
$$[\ a,\ b,\ b,\ c,\ f,\ f,\ e,\ a,\ a,\ d,\ f,\ e\ ]\}$$

$$B = \{[\ c,\ d,\ f,\ a,\ c,\ d,\ f,\ a,\ c,\ d,\ f,\ a\ ],$$
$$[\ c,\ d,\ f,\ a,\ c,\ d,\ f,\ a,\ c,\ d,\ f,\ f,\ f\ ],$$
$$[\ c,\ d,\ f,\ a,\ c,\ d,\ f,\ a\ ]\}$$

The question is, can we characterize the difference between random and patterned behavior of this robot?

We will represent our hypotheses as deterministic finite state automata(FSA). Even with this restricted representation scheme finding a globally optimal encoding is not feasible because the range of possible hypotheses grows exponentially with the amount of data in the sample.

At one extreme we can generate a machine which accepts only previously seen examples as instances of the concept. This can be encoded as a deterministic FSA that accepts exactly the examples seen[Figure 4 A, B]. At the other extreme we can encode concepts with a universal acceptor which simply accepts any string at all [Figure 4 A'].

The goal here is to find the optimal point between these two extremes that generalizes beyond the observed data but not too far. To this end we propose the local search algorithm presented in Table 1, which finds an approximately minimum description length. It is similar, but not identical, to that used earlier by Teal et al.[29].
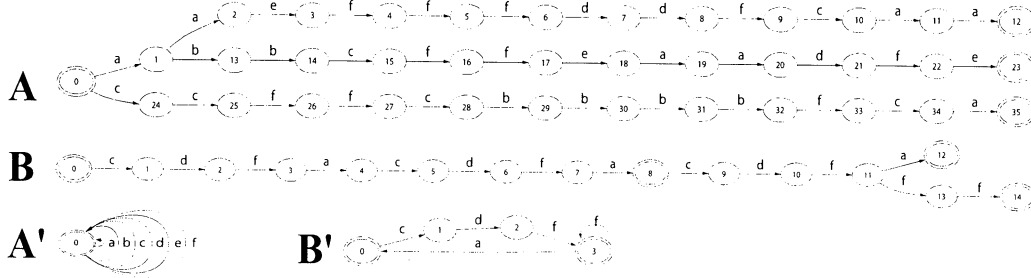
Figure 4: **A**. Prefix tree acceptor derived from data set A. **B**. Prefix tree acceptor derived from data set B. **A'**. MDL machine found for data set A. **B'**. MDL machine found for data set B.

Table 1: **Local Search Algorithm**

Step	Task
1	Build a prefix tree acceptor for the observed strings.
2	Try merging each pair of nodes in the machine keeping only new machines for which the cost of the machine + the cost of encoding the strings is cheaper than the best seen so far.
3	Repeat step 2 on the new machines until no single merger yields a cheaper machine + encoding pair.

Table 2: **MDL Encoding Costs**

		Set A	Set B
Initial	Figure 4	A	B
cost of encoding machine		467.9	147.2
cost of encoding messages		5	3
Total cost		472.9	148.2
Final	Figure 4	A'	B'
cost of encoding machine		15.5	34
cost of encoding messages		101.1	22.4
Total cost		116.6	56.3

Following this algorithm, an initial prefix tree acceptors for set A and B are constructed as Figure 4A and Figure 4B. The size of a given hypothesis of the data can be calculated as the cost of the binary encoding of the machine plus the cost of encoding the observed stimuli in that machine. The cost of encoding a given deterministic FSA is defined as the following:

$$|d|\bigl(2\log_2 |Q| + \log_2 |S|\bigr) + |F|\log_2 |Q|$$

where $|Q|$ is the number of nodes in the machine, $|S|$ is the number of symbols used, $|d|$ is the number of arcs in the machine and $|F|$ is the number of final states.

The cost of encoding a message in a given deterministic FSA is :

$$\sum_{i=1}^{m}\sum_{j=1}^{|S_i|} \log_2 Z_{i,j}$$

Where m is the number of sentences in the sequence of strings encoded, $|S_i|$ is the length of the ith string s_i, and $z_{i,j}$ is the number of ways to exit the state reached on the ith symbol of the string s_i. This is just one(particularly simple) metric of the size of a given encoding. The local search algorithm described in Table 1 reduces the cost of encoding a message.

3.3 Results of Local-search MDL

Figure 4 A' and B' are the final deterministic FSA resulting from the analysis of example data sets A and B. The cost of each FSA and encoding are compared in Table 2.

The machine found for data set A (Figure 4A') encodes the strings from set B less efficiently than the machine found for data set B (Figure 4B'); a cost of 92.6 for the messages, yielding a total cost of 108.5 compared to a total cost of 56.3 for B'. The machine found for data set B won't recognize strings from data set A at all. This algorithm cor-

rectly distinguishes the random behavior in data set A from the patterned behavior in data set B. Finally we should note that while this metric of description length was effective in discriminating simple patterns in the data for robot movement behavior, different domains will require different metrics of generalization.

4 Discussion and Conclusion

The framework provided by viewing collaboration of heterogeneous agents as language acquisition provides a well explored theoretical basis for many of the most vexing problems. As a first step in implementing such a language acquisition system, we have bound real-world sensor data to symbols using both a classification system for object grounding and an MDL approach for event grounding.

Within the scope of auditory data from frog calls a simple classification system has been implemented and provides a general model that can be easily extended. The combination of a relatively easy data source combined with the fact that the linguistic grounding process can tolerate some misclassification errors motivates more sophisticated work on good, but not perfect, classifiers.

The MDL strategy presented above is effective for distinguishing among some simple patterns of sensory data corresponding to different types of events. The next logical step in attempting to ground events is to develop strategies for more complex events and noisier data. To this end we have begun preliminary investigations using MDL techniques on ethograms.

Originally devised by naturalists studying animal behavior, ethograms catalog the behavioral events (or specific action patterns) that animals use during different behavioral states or contexts. Since ethograms can be encoded as series of behavioral events, it seems plausible to extract a generalized pattern of behavior of a given animal from its ethogram using MDL method.

Our initial examinations of marmot ethograms (obtained from Dr. David Blumstein at UCLA) using the MDL method presented above did not yield any patterns that could be encoded with deterministic finite automata other than a universal acceptor. This serves to illustrate the crucial relationship between the representation scheme used to formulate the hypotheses and the concepts that are the target of generalization. For future investigation of ethograms and other noisy patterns probabilistic nondeterministic finite automata may prove more effective. We are currently exploring this possibility.

Acknowledgments

References

[1] K. Wee, T. Collier, G. Kobele, E. Stabler, and C. Taylor. Natural language interface to an intrusion detection system. In *Proceedings, International Conference on Control, Automation and Systems*. ICCAS, 2001.

[2] S. Harnad. The symbol grounding problem. *Physica D*, 42:335 346, 1990.

[3] J. M. Siskind. A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition*, 61(1-2):39–91, 1996.

[4] E. M. Gold. Language identification in the limit. *Information and Control*, 10:447–474, 1967.

[5] D. Angluin. Inference of reversible languages. *Journal of the Association for Computing Machinery*, 29:741–765, 1982.

[6] M. Kanazawa. Identification in the limit of categorial grammars. *Journal of Logic, Language, and Information*, 5:115–155, 1996.

[7] F. Denis. Learning regular languages from simple positive examples. *Machine Learning*, 44:37–66, 2001.

[8] E. P. Stabler. Acquiring grammars with movement. *Syntax*, 1:72 97, 1998.

[9] E. P. Stabler. Identifying minimalist languages from dependency structures. Presentation available at http://www.clsp.jhu.edu/seminars/abstracts/S2002/stabler.shtml, 2002.

[10] N. Komarova, P. Niyogi, and M. Nowak. Evolutionary dynamics of grammar acquisition. *Journal of Theoretical Biology*, 209(1):43–59, 2001.

[11] L. Steels. Self-organising vocabularies. *IEEE Intelligent systems*, pages 16 23, 2001.

[12] S. Kirby. Spontaneous evolution of linguistic structure: an iterated learning model of the emergence of regularity and irregularity. *IEEE Transactions on Evolutionary Computation*, 5:102–110, 2001.

[13] M. Oliphant. The dilemma of saussurean communication. *BioSystems*, 37:31–38, 1996.

[14] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, NY, 2000.

[15] P. Vitányi and M. Li. On prediction by data compression. In *Proceedings of the 9th European Conference on Machine Learning*, Lecture Notes in Artificial Intelligence, Vol. 1224, pages 14–30, 1997.

[16] P. Vitányi and M. Li. Minimum description length induction, Bayesianism, and Kolmogorov complexity. *IEEE Transactions on Information Theory*, IT-46:446–464, 2000.

[17] H. Wang, J. Elson, L. Girod, D. Estrin, and K. Yao. Target classification and localization in a habitat-mornitoring application. submitted to ICASSP 2003, Hong Kong, 2003.

[18] H. Wang, L. Yip, D. Maniezzo, J. C. Chen, R. E. Hudson, J. Elseon, and K. Yao. A wireless time-synchronized cots sensor platform part ii: Applications to beamforming. In *In Proceedings of the IEEE CAS Workshop on Wireless Communications and Networking, Pasadena, California, September 5-6, 2002*. IEEE, 2002.

[19] C. Marty and P. Gaucher. Guide sonore des amphibiens anoures de guyane. Compact-Disc, 2002.

[20] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Englewood Cliffs NJ: PTR Prentice Hall (Signal Processing Series), 1993. General Intro : ISBN 0-13-015157-2.

[21] D. Mumford. Pattern theory: a unifying perspective. In D. C. Knill and W. Richards, editors, *Perception as Bayesian Inference*. CUP, NY, 1996.

[22] M. R. Brent and T. A. Cartwright. Lexical categorization: Fitting template grammars by incremental MDL optimization. In L. Micla and C. de la Higuera, editors, *Grammatical Inference: Learning Syntax from Sentences*, pages 84–94. Springer, NY, 1996.

[23] J. Rissanen and E. Ristad. Language acquistion in the MDL framework. In E. Ristad, editor, *Language Computations*. American Mathematical Society, Philadelphia, 1994.

[24] P. Grünwald. A minimum description length approach to grammar inference. In G. S. S. Wermter, E. Riloff, editor, *Symbolic, Connectionist and Statistical Approaches to Learning for Natural Language Processing*, LNCS #1040. Springer-Verlag, Berlin, 1996.

[25] J. Goldsmith. Unsupervised learning of the morphology of a natural language. *CL*, 27(2):153–198, 2001.

[26] T. K. Teal, D. Albro, E. Stabler, and C. E. Taylor. Compression and adaptation. In *European Conference on Artificial Life, ECAL '99*, 1999.

[27] A. Kolmogorov. Three approaches to the definition of the concept 'amount of information'. *Selected Translations in Mathematical Statistics and Probability*, 7:293–302, 1968.

[28] A. N. Soklakov. Complexity analysis for algorithmically simple strings. *arXiv:cs.LG*, 0009001, 2001.

[29] T. K. Teal and C. E. Taylor. Effects of acquisition on language evolution. *Artificial Life*, 6(2):129–143, Spring 2000. als titled as: Effects of Compression on Language Evolution.

Real-time face tracker using ellipse fitting and color look-up table in irregular illumination

Hyun Seok Hong   Dong Hyun Yoo   Myung Jin Chung

Department of Electrical Engineering and Computer Science, KAIST
373-1, Guseong-dong,
Yuseong-gu, Daejeon, 305-701, Korea

wiser@cheonji.kaist.ac.kr   ydh@cheonji.kaist.ac.kr   mjchung@ee.kaist.ac.kr

Abstract

•• In this paper, a real-time face tracker for a service robot is introduced. Color information is very useful for detecting human skin color, and makes it possible to reduce the searching area and searching time. We use H and S value of HSI color model to detect human skin color. To cope with illumination change during tracking in real-time, we use a color look-up-table, which is made in all illumination change. Region grouping process is applied to color segmented image to find face candidate blobs. At each frame, we apply pattern matching to every face candidate blob using normalized correlation coefficients to verify face pattern. Each face candidate blob is fitted by an ellipse, and its major and minor axis are computed. The direction of the major axis determines the planar rotation angle of a face. Furthermore, the length of the minor axis determines the size of the face template. This method makes the face detection fast and detect 2D rotation angle. To achieve high reliability of face detection, we use light condition compensation and histogram equalization as preprocess of pattern matching. This real-time face tracker is efficient for human-robot interaction, e.g. face recognition and eye-gaze tracking system.

1. Introduction

The work described in this paper is part of a human-friendly care robot system for the elderly. The main goal of the care robot is to support walking of an old and feeble person. The robot has three functions that are sound localization, face tracking and autonomous navigation. When the care robot is far from a user, he calls it by clapping his hand. Then, it detects the relative direction of the user and turns its head toward him. If the robot finds a human face, it starts face tracking and comes to him using autonomous navigation. To track a human face robustly, the tracker should cope with an uneven lighting condition and illumination change, and check whether the tracking object is a human face or not. Also, the tracker should work in real-time and needs to track only one person who asks for services.

Yang and Waibel proposed face tracking algorithm which is based on skin color filtering [1]. Skin color is distinguisthed by a Gaussian color model in normalized-RG color space, and the color model is updated by accumulating Gaussian parameters of previous detected face. However, this algorithm cannot assure whether the segmented color region presents a face or not. Furthermore, the updated color model may not be converged to human skin values due to the background colors similar to the skin color. McKenna and Gong proposed a face tracking algorithm which is based on motion detection and front-view face detection [2]. This algorithm cannot be used in an active camera system because moving region detection is hindered by background motion. Lee et al. [3] proposed a color modeling approach in HSI color space using B-spline curves based on the fact that the color distribution of a single colored object in HS plane has variation in irregular illumination. However, this method also does not assure whether a segmented color region presents a face or not. Terrillon et al. used the eleven lowest-order moments of each segmented color segment blob as the input of feed-forward multilayer perceptron [4]. However, it does not use any facial feature, so it may detect other body parts (face including neck, hand) and other objects as faces. Also there is no color model adaptation to changing illumination. Rowley et al. proposed neural network based face detection algorithm [5]. This algorithm can cope with uneven illumination on face. However, it tries to find many different sizes of faces in an image without knowing face size. Thus, it is not adequate for a real-time face tracker.

In this paper, we propose a new method for face tracking to deal with irregular illumination variation, uneven lighting condition, planar rotated frontal faces verification and detection of rotation angle in real-time.

Because the user and the mobile robot should not be constrained to fixed positions, we use a pan-tilt camera which tracks the user while he moves towards the camera and tries to keep the tracked face always centered in the image.

We use a color look-up-table to find human skin region fast. Then each skin blob is acquired by region-

grouping and analyzed by the second order moments. As the result, blob size and planar rotation angle, the minor axis length and the major axis direction of each blob are acquired and some of the blobs are classified into face candidates. For each face candidates, uneven lighting compensation and histogram equalization are processed as preprocesses of pattern matching. Pattern matching verifies whether the blob is a face or not.

Face candidate extraction is introduced in chapter 2. In chapter 3, we explain a fast face verification method using face candidate information. Experimental results are shown in chapter 4.

2. Face candidate extraction

2.1. Color segmentation

The pixels of an image are compared with a specific color model and segmented by thresholds in color space. There are many kinds of color coordinate systems, RGB, HSI, HSV, YUV, YC_bC_r, YIQ, UVW, $L^*a^*b^*$, etc. Each color coordinate system has its own property [6]. We use the HSI color coordinate system for skin color detection because hue and saturation values are invariant to intensity change. The skin color model is acquired in the form of Gaussian function in hue and saturation plane with respect to each intensity value. We use a look-up-table for fast color segmentation. Furthermore, to cope with illumination change, we use finite number of look-up-tables according to quantized intensity value of estimated face region. Therefore, skin color model adaptation is processed according to illumination of face region. Figure 1 shows acquisition of color look-up-tables of each face region intensity range.
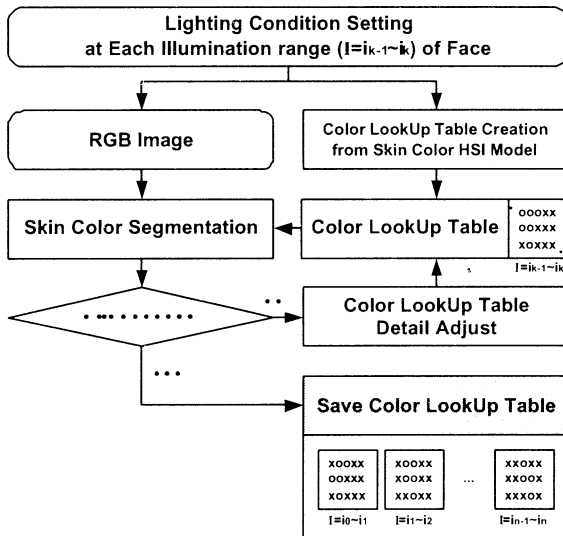
Figure 1. Skin color look-up-table acquisition algorithm

2.2. Face candidate extraction

Region grouping distinguishes each connected

components in the binary image, which is acquired by color segmentation. Each region is regarded as a face candidate region.

By intuition, a human face can be modeled by an ellipse. The length of the minor axis is proportional to face width. Also the direction of the major axis means the planar rotation angle of face for frontal face.

Ellipse fitting of each region can be calculated by the second order moments, which are following.

$$M = \begin{bmatrix} \overline{m}_{20} & \overline{m}_{11} \\ \overline{m}_{11} & \overline{m}_{02} \end{bmatrix} \tag{1}$$

$$\overline{m}_{11} = \sum_x \sum_y (x - \overline{x})(y - \overline{y}) \tag{2}$$

$$\overline{m}_{20} = \sum_x \sum_x (x - \overline{x})^2 \tag{3}$$

$$\overline{m}_{02} = \sum_y \sum_y (y - \overline{y})^2 \tag{4}$$

$$\overline{x} = mean(x) \tag{5}$$

$$\overline{y} = mean(y) \tag{6}$$

The bigger eigen value of M and corresponding eigen vector present the length and the direction of the major axis respectively. In the same way, another eigen value and vector mean the length and the direction of the minor axis.

In a solid ellipse, the actual length of the minor axis is four times the smaller eigen value. Figure 2 shows ellipse fitting result. a_i and b_i are the half length vectors of the minor axis and the major axis, and θ_i means detected planar face rotation angle ($i = 1,2,3.$).

In the normal situation, the ratio between the major axis length and the minor axis length of a human face is limited in some range. Therefore, we can exclude the blobs which exceed a specific ratio value from face candidates.
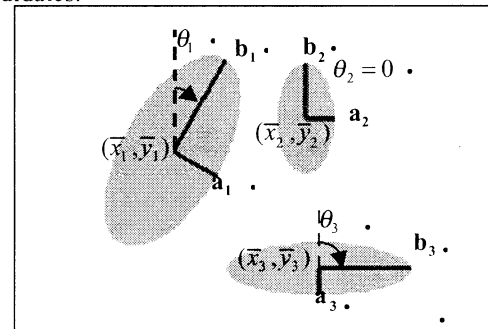
Figure 2. Ellipse fitting using moments

3. Face pattern-matching

3.1. Face width and orientation

The Face tracker, which does not use any facial

features, cannot guarantee that it is really tracking the human face. However feature-matching process takes generally much calculation time. To diminish the time, we fix face template size to 20 × 20 pixels. When the region of a face candidate is bigger than template's size (20 × 20 pixels), the region is sub-sampled to the size of face template.

Each face candidate contains information about the region including its center position $(\overline{x}_i, \overline{y}_i)$, planar rotation angle (θ_i) and face template width (w) where $i = 1, \cdots, N$ and N is the number of face candidates. We make face pattern α times as long as the length of the minor axis($\lfloor 2a_i \rfloor$).

Following preprocess and face pattern-matching are processed in the small neighborhood of center position, in the direction of planar rotation angle, with face template's width specified by each face candidate information. By doing this, the search area is reduced and the number of matchings is remarkably reduced using a fixed template size. This algorithm is shown in Figure 3.
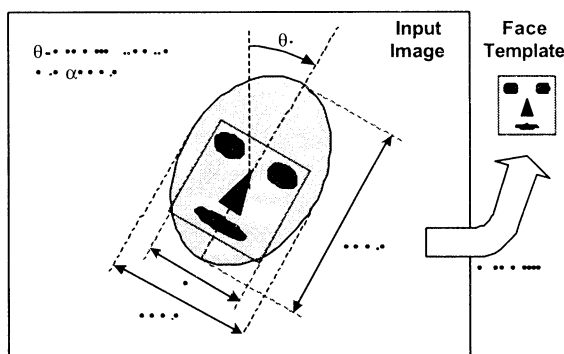
Figure 3. Finding the width and the orientation of face candidate for matching with face template.

3.2. Preprocess

Preprocess for the face pattern-matching is composed of two stages. For the first step, uneven lighting is compensated by 2D linearization. Then the gray-level image of each face candidate is equalized. The lighting compensation reduces the effect of non-uniform lighting by subtracting 2D linear fitted image from the original gray image. Then, histogram equalization emphasizes the features on gray image [5]. These preprocess makes the pattern-matching result more reliable.

Figure 4 shows how this preprocess works on uneven and even illumination environment.

3.3. Pattern-matching

In this step, pattern-matching is carried out to determine whether face candidates are true faces or not. Pattern matching between face template and face candidate is executed by inspecting normalized correlation coefficient.
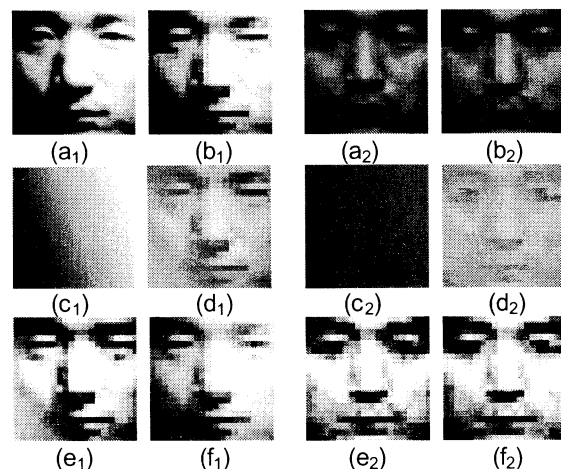
Figure 4. (a₁)~(f₁): uneven lighting condition case, (a₂)~(f₂): even lighting condition case; (a₁), (a₂): Selected region in face candidate, (b₁), (b₂): sampled image, (c₁), (c₂): Linear fitted illumination, (d₁), (d₂): uneven illumination compensation, (e₁), (e₂): histogram equalized image of (d₁), (d₂) respectively, (f₁), (f₂): histogram equalized image of (b₁), (b₂) respectively.

Figure 5. Partial face images for creating face pattern and created face template.

Figure 5 shows some of database images from Yale Face Database to make the face template. Our face model is the average image of the histogram equalized images in the database.

We regard the face candidate as a face if the correlation coefficient is greater than a certain threshold value. The correlation coefficient is defined in -1 to 1.

4. Experiment
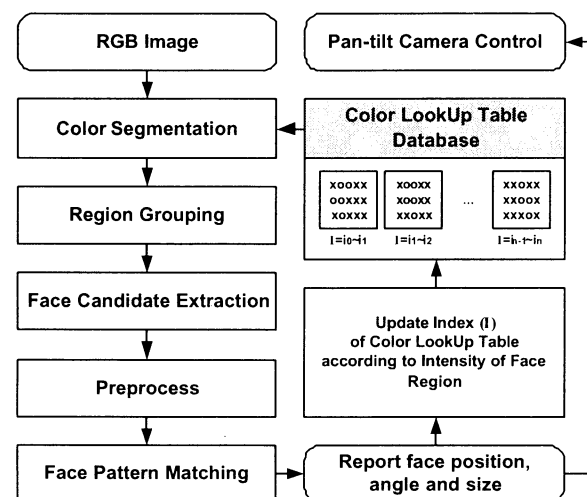
Figure 6. Overall Algorithm

Overall algorithm is drawn in Figure 6.

In pattern matching, we set $w = 0.75 \times |2a_i|$ by observing average human face.
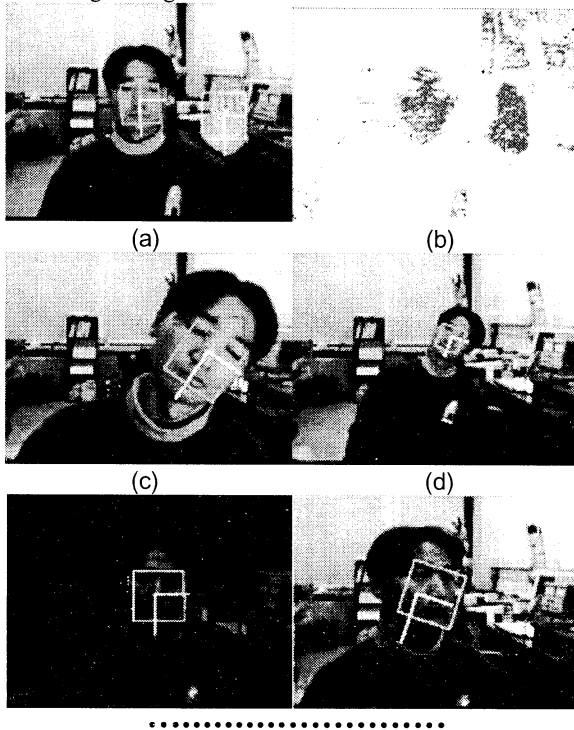
Figure 7. Face tracking result; (a) tracking only face, (b) color segmented result, (c) rotated big face, (d) rotated small face, (e) illumination changes, (f) 3D rotation detected

Figure7 shows that our face tracker detects only human face, regardless of its size. Also when illumination changes it tracks face robustly. Figure7.(f) shows 3D rotated face. It is detected but its 3D rotation angle is not distinguished from planar rotation angle.

Stage	Elapsed Time
Color Segmentation	1~2ms
Region Grouping	<<1ms
Face Candidates Extraction	<<1ms
Preprocess +Face Pattern Matching	12~56ms
Total	13~60ms

Table 1. Elapsed Time

	|Error| [deg]	Standard deviation
even illumination	2.5657	1.5946
uneven illumination	5.0124	3.2303

Table 2. Face rotation error

Elapsed time is indicated in Table 1. one frame is processed in 60msec at least. Table 2 shows detected angle error. Uneven illumination makes the direction of the color segmented blob inaccurate.

5. Conclusion.

We introduced a real-time face tracker, which verifies human face at each time. It tracks human face, and detect the face feature region and planar rotational angle in real-time, even though there is uneven lighting in the environment. However, in case that the skin color objects are stuck to face in the direction of width, it can track the face but cannot verify the face. Our algorithm can be applied to face recognition system that extracts face feature region regardless of planar rotation. Also, as a human-machine interface, some information exchange between a human and a robot can be done by our face tracker.

References

[1] J. Yang and A. Waibel, "Tracking human faces in real-time", Proc. IEEE Workshop on Applications of Computer Vision, 1996.

[2] S. J. McKenna and S. Gong, "Tracking faces", Proc. International Conference on Face and Gesture Recognition, pp. 271-276, 1996.

[3] Yong-Beom Lee, Bum-Jae You, Seong-Whan Lee, "A real-time color-based object tracking robust to irregular illumination variation.", Proceedings of the 2001 IEEE International Conference on Robotics and Automation, pp.1659 -1664 vol.2 , 2001

[4] Terrillon, J.-C.; David, M.; Akamatsu, S., "Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments", Proceedings. Third IEEE International Conference on Automatic Face and Gesture Recognition, pp.112 -117, 1998.

[5] Rowley, H.A.; Baluja, S.; Kanade, T., "Neural network-based face detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.23 -38, Volume: 20 Issue: 1 , Jan. 1998.

[6] Zarit, B.D.; Super, B.J.; Quek, F.K.H., "Comparison of five color models in skin pixel classification", Proceedings. International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, pp.58-63, 1999.

Fast Face Detection Using Genetic Algorithms and Pyramid Structure

Masanori Sugisaka[1,2], and Xinjian Fan[1]
[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

Hidenori Kimura[2,3]
[2]The Institute of Physical and Chemical Research
(RIKEN), Bio-Mimetric Control Research Center,
Shimoshidami, Moriyama-ku Nagoya, 463-0003, Japan
[3]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: Face detection is the fundamental step before face recognition or identification procedure. Its time-response has a major influence on the performance and usability of the whole face recognition system. However, most of existed face detection methods suffer from a slow time-response. In this paper, we proposed to achieve fast face detection by using genetic algorithms (GAs). Specifically, we used GAs to search for sub-images that might contain a face: each individual in the population represents a sub-image within the input image; to evaluate each sub-image, we use its matching value against a neural network based face model. Different from other genetic search methods, our approach works on an image pyramid, which makes it more computationally feasible.

Keywords: Face detection, neural network, genetic algorithms, image pyramid

1 Introduction

Face detection is the fundamental step before face recognition or identification procedure. Its reliability and time-response have a major influence on the performance and usability of the whole face recognition system. Given a single image or a sequence of images, the goal of the face detection is to return a list of locations which correspond to human faces. The base procedure of face detection is shown in Figure 1.

In Figure 1, F is the face model and T is the input image. The task is to find a region in the input image T that is best matched with the face model. Here we make use of the constraint that the height and width of the face in the input image are no less than those of the face model. Let F_s represent a current candidate region. It should first be scaled down to the size of the face model and then calculated how well it matches the model. Let's denote the matching value between F_s and F as MV. With this notation the face detection problem can be stated as:

$$\max_{F_s \in T} MV(F_s, F) \qquad (1)$$

If $MV(F_s, F)$ is higher than a preset threshold, the corresponding portion of F_s is declared as face candidate.
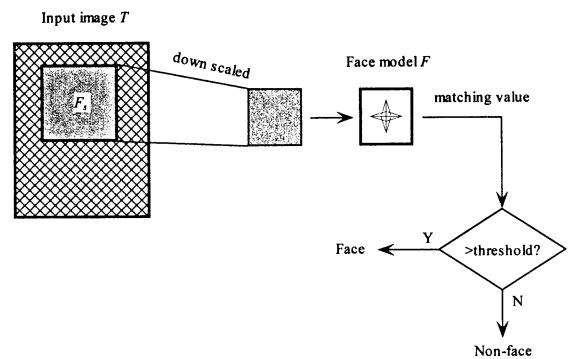
Fig.1 Basic face detection procedure

Up to now many face detection methods have been published. According to how the face model is built and how the input image is searched, the methods can be categorized as follows:

- Methods based on facial features[1][2].
- Methods based on face representations learned from a large number of examples (face images) using statistical approaches (e.g., eigenfaces)[3] or neural networks[4][5].

In general, methods in the second category are more practical since they do not rely on special features.

- Methods using an exhaustive search: a sliding window moves one pixel by one pixel along left to right and then down top to bottom. To detect faces larger than the window size, the input image is repeatedly reduced in size by a factor, and the detector is applied at each scale. This is the way used in most face detection systems. The exhaustive search would produce the optimal result, but due to the exponential complexity it is not computationally feasible.
- Methods using an optimal search. The task of finding a well-suited sub-image can be formulated as a discrete global optimization problem. Search methods based on some optimal algorithms, such as dynamic programming (DP)[6] and genetic algorithms[7][8] have been proposed. Among of them, GAs are adaptive search procedures inspired by the mechanics of natural evolution and have shown to

perform well when considering large search spaces such as the one considered in this work.

Our particular system is based on neural network and genetic algorithm: neural network serves as a face model while genetic algorithm is used to search the image efficiently. Unlike the work described in [7] and [8], our genetic search works on an image pyramid, which make it more computationally feasible.

Here face detection is made on grey-level images without using color information. Faces are vertically oriented frontal views or slightly rotated views. Wide face expression changes are also allowed. For simplicity, we concentrate on the task of finding a single face in an image. The extension to finding multiple faces is straightforward.

2 Neural network based face model

The face model should grasp as much as possible the common features of human face, while as the same time is not vulnerable to the background and individual feature. Our face model is based on neural network, which is much like the work described in [4] and [5]. The neural network is trained with many face examples images to learn the concept of face. It receives a 20×20 pixel region of the image as input, and generates an output ranging from 0 to 1, signing how close it is to a face.

Obtaining a sufficient number of examples is an important problem for face detection. Out face examples are gotten from the Internet face database, which are all of up-right frontal type. The training database featured also slightly rotated faces. This improves the robustness of face detection process. The face images contained in the training set are aligned to have a 20×20 size and preprocessed using the method described in the following section. Additional non-faces were introduced by applying the bootstrap algorithm.

2.3 Image preprocessing

To reduce variation caused by lighting or camera differences, after a sub-image is extracted from a particular location, it is preprocessed with standard algorithms such as histogram equalization to improve the overall brightness and contrast in the images. The preprocessed window is then passed to the trained neural network.

2.4 Neural network output visualization

Figure 3 is shown a 3D plot of the raw output of a neural network obtained with the image on the left hand side as input. The high needle-like peak represents face location. Outside the peak are wide open spaces featuring very low values.

Fig.2 Neural network based face model

(a) Image "face"   (b) NN output for "face", 3D view

Fig.3 Neural network output visualization

2.1 Neural network

We consider a 3-layer feed-forward neural network to serve as a face model. The network (see Figure 2) trained with the classical back-propagation algorithm. The input layer is a vector, and the output layer is a single neuron. We consider $n \times n$ pixel input images transformed into a column-vector to feed the input layer constituted by n^2 units or neurons. For the work presented here, $n = 20$.

2.2 Training data

The training data is a set of selected patches containing faces (positive examples) and random patches chosen from an image containing no faces (negative examples).

2.5 Threshold setting

The sub-image with a matching value above the threshold would be regarded as a face. Setting correctly the output threshold is of great importance. At a threshold of 1, the false detection rate is zero, but few faces are detected. As the threshold decreased, the number of correct detections will increase, but so will the number of false detections. The setting of threshold also depends on the network currently applied. One can only estimate the appropriate threshold by doing some test runs using a variety of thresholds. In our work, a threshold of 0.9 is used.

3 Methodology

In the following sections, we will discuss the steps of the proposed approach in detail.

3.1 Image pyramid

Genetic search for face detection was also used in [7] and [8]. In their methods, the algorithm started from a population of sub-images randomly abstracted from the original input image. Each sub-image is evaluated according to how well it matches an Eigenface based[7] or a template based[8] face model. To evaluate sub-images of different sizes, they scaled them down to the size of the face model. However the above procedure is too time consuming that we doubt their methods' usability. In this paper we proposed to enhance the genetic search method by using a pyramid structure.

The idea of using pyramid structures in image analysis was introduced by Tanimoto and Pavlidis[9] as a solution to edge detection. Since the 1970's, considerable research activities have been taken place in this area and image pyramids are now used in several areas of image analysis. One important property of the pyramid structure is that it is computationally extremely efficient[10].

The image pyramid is generated by scaling down the original input image with a predefined factor multiples times. The top level (the coarsest resolution image) should have a size equal to or greater than that of the face model. And the number of pyramid levels must be large enough otherwise some faces could fail to be detected because they fall within scale regions that are not covered by the pyramid. Moreover, a face can be detected at multiple nearby positions and scales. The number of nearby detections has a large effect on genetic search: the larger the number, the more easily for the GA to find the face. Too few scales would result in too few nearby detections that would make the genetic search very difficult. In our work, a scale factor of 1.2 is used, resulting in a 14-level pyramid for a 320×240 input image.

3.2 Representation

In our genetic method, sub-images (individuals) have a fixed size of 20×20, extracted from different levels of the pyramid. To define a sub-image, we used it mapped center (C_x, C_y) at the original image (*Level* zero) and the scale level s which it lies in. So the chromosome of an individual consists of 3 genes, represented by the data structure shown in Figure 4.

The relationship between the center of the search window $(C_x^{(s)}, C_y^{(s)})$ and its mapped point at the original image (C_x, C_y) is stated as follow:

$$C_x^{(s)} = \frac{C_x}{q^s}, C_y^{(s)} = \frac{C_y}{q^s} \qquad (2)$$

where q is the scaling factor. The sub-window should
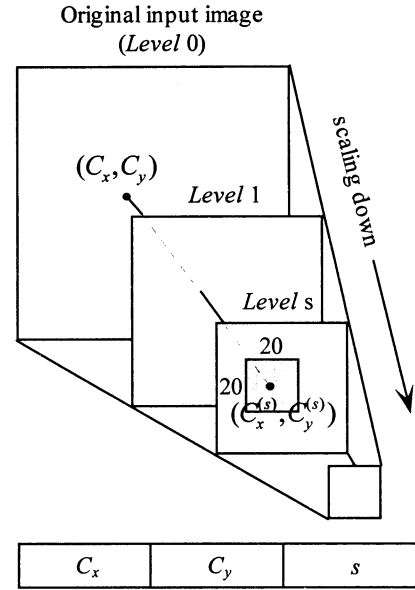
Fig.4 Chromosome representation

lie within the image, so

$$10 \leq C_x^{(s)} \leq M^{(s)} - 10$$
$$10 \leq C_y^{(s)} \leq N^{(s)} - 10 \qquad (3)$$

where $(M^{(s)}, N^{(s)})$ is the size of *Level s*. Let (M, N) be the size of the original image, then we have $M^{(s)} = \frac{M}{q^s}, N^{(s)} = \frac{N}{q^s}$.

3.3 Genetic design

A real-coded GA (RGA) is used here. Therefore, the coding and decoding processes that are needed in the binary-coded GA are avoided, thus increasing the GA's speed.

(1) Crossover operator

Let us assume that $\overline{P_1}$ and $\overline{P_2}$ are two chromosomes that have been selected to apply the crossover to them. Two offspring $\overline{P_1'}$ and $\overline{P_2'}$ are generated,

$$\overline{P_1'} = \lambda \overline{P_1} + (1-\lambda)\overline{P_2}$$
$$\overline{P_2'} = (1-\lambda)\overline{P_1} + \lambda \overline{P_2} \qquad (4)$$

where λ is an uniform random number in [0, 1].

(2) Mutation operator

Let us suppose $\overline{C} = (c_1, \cdots, c_2, \cdots, c_n)$ a chromosome and $c_i \in [a_i, b_i]$ a gene to be mutated. Next, the gene, c_i', resulting from the application of mutation operators.

$$c_i' = c_i + \gamma \cdot dev_i \qquad (5)$$

where dev_i defines the mutation range and it is normally set to $0.1 \cdot (b_i - a_i)$. γ is randomly generated from [-1, 1].

Other operators and parameters for RGA were set as

summarized in Table 1.

Table 1 Setting for RGA

Operator/Parameter	Type/Value
Selection	Roulette wheel selection with elitism
Fitness scaling	Linear
Size of population	70
Number of generations	100
Probability of crossover	0.90
Probability of mutation	0.30
Generation gap	1.0

3.4 Fitness evaluation

To compute a sub-image's fitness, we directly use it matching value against the neural network based face model. The larger the value, the more the sub-image resembles a face.

4 Experiments and conclusion

Using the techniques listed above, we have built a real-time system which continuously gets image from a CCD camera, finds face in it. If a face is detected, the system will mark it with a square, otherwise it will say "no face found". The system was run on a 422 MHz Pentium® II processor, 128 MB memory, using an image size of 320×240 at about 2 frames per second. The system was developed with Visual C++ 6.0 under Windows 98 SE. We have tested the above algorithm on many scenes of an office environment with different illumination conditions and increasing complexity background. Here we show one of them (Figure 5). The algorithm stopped at the 100th generation and the results reported were based on 15 runs. On the average, 20 generations were required for the algorithm to find the face. The sub-images explored by the GA are much less than by performing an exhaustive search (See Table 2).

References

[1] Huang C. and Chen C. Human, "Facial feature extraction for face interpretation and recognition", Pattern Recognition, 25(12): 1435-1444, 1992

[2] Brunelli R. and Poggio T., "Face recognition: Features versus templates", IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(10): 1042-1052, 1993

[3] Turk.M. and Pentland A., "Eigenfaces for recognition", Journal of Cognitive Neuroscience, 3: 71-86, 1991

[4] Henry A. Rowley, "Neural Network-Based Face Detection", Thesis submitted for the degree of Doctor of
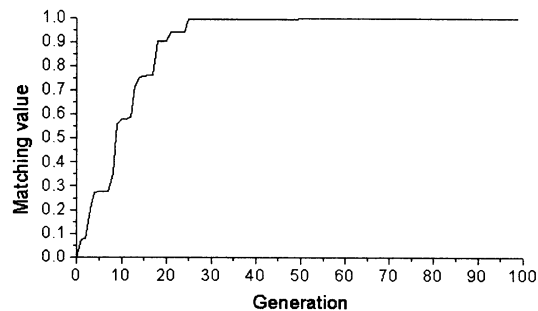
Fig.5 A scene and its GA performance plot

Table 2 Comparison

	Proposed approach	Exhaustive search in the same image pyramid
Number of searched sub-images	1,400	197,088

Philosophy, School of Computer Science, Carnegie Mellon University, 1999

[5] B. Fasel, "Fast multi-scale face detection", IDIAP-COM 98-04, IDIAP, 1998

[6] Z. Liu and Y. Wang, "Face detection and tracking in video using dynamic programming", Proceedings of the 2000 International Conference on Image Processing, 2000

[7] George Bebis and Satishkumar Uthiram, "Face Detection and Verification Using Genetic Search", International Journal on Artificial Intelligence Tools, Vol. 9, No. 2 (2000) pp. 225-246

[8] Y.Yokoo & M.Hagiwara, "Human Face Detection Method Using Genetic Algorithm", Transactions of The Institute of Electrical Engineers of Japan, Vol. 117-C, pp. 1245-1251, 1997

[9] S. Tanimoto, T. Pavlidis, "A Hierarchical Data Structure for Picture Processing", Computer Graphics and Image Processing, Vol. 4, pp. 104-119, 1975

[10] W.G. Kropatsch, "Properties of Pyramidal Representations", Computing Suppl. 11, pp. 99-111, 1996

A study on color-based line tracking

Masanori Sugisaka, Renping Chen

Department of Electrical and Electronics Engineering
Oita University, 700 Dannoharu, Oita 870-1192, Japan
Email: msgi@cc.oita-u.ac.jp chenrp@cc.oita-u.ac.jp

Abstract: For a mobile robot vision system, the color based detection and track technique is more efficient than others. But its reliability can be strongly affected by the lighting conditions in the environment. In order to reduce the effect of different lighting conditions when line tracking, luminance must be removed from line color representations and detected images. In this paper, we will present a simple and efficient method that color-based line tracking. In our case, we first convert the RGB images to HSV (Hue, Saturation, Value) images, and then only use the hue value to build the line colour model. The line colour model will used to detect line colour regions from non-line colour regions in images taken by a CCD video camera. At last, the COG (center of gravity) of the line color distribution will be computed to track line.

Key works: Color model, HSV color space, COG

1 Introduction

For a mobile robot vision system, the color-based line tracking algorithms must be fast and efficient. It must be able to track in real time yet not absorb a major share of computational resources: other tasks must be able to run while the vision system is being used. But, if the color message is built by common RGB color space, its reliability can be strongly affected by the lighting conditions in the environment. This is because the triple components (R, G, B) in a common RGB image represents not only color but also luminance. Luminance will vary under different lighting conditions [2][4]. In order to reduce the effect of different lighting conditions when line tracking, luminance must be removed from line color representations and detected images. In our case, in order to solve this problem, we first convert the RGB images to HSV (Hue, Saturation, Value) images, and then use the hue value to build the line color model. The line color model will used to detect line color regions from non-line color regions in images taken by a CCD video camera. At last, the COG (center of gravity) of the line color distribution will be computed to track line.

2. Line detection

In our case, line is with special color, and the robot tracks this line by its color to move, so we first must to detect line region from nonline regions. In practical applications, the robot will often work under different lighting conditions. If it was detected only based on a group of special RGB values of the line, the detection results are not always good, and sometimes make the line segmentation more difficult [3]. In order to solve

this problem, the luminance message must be removed from line color representations. In order to remove luminance message, we built the line color model only by its hue value distribution and detect the line region by its color model in the HSV color space. The process to build line color model is as follows:

① Extract the line region from common RGB color images under different lighting conditions to constitute a sample image about the line color.

② Transform each pixel color in this sample image from the RGB color space to the HSV color space by next formulas.

$$H = \begin{cases} 1 - \dfrac{R-G}{\max(R,G)-B} & when \quad B = \min(R,G,B) \\ 3 - \dfrac{G-B}{\max(G,B)-R} & when \quad R = \min(R,G,B) \\ 5 - \dfrac{B-R}{\max(B,R)-G} & when \quad G = \min(R,G,B) \end{cases}$$

$$S = 1 - \frac{\min(R,G,B)}{\max(R,G,B)}$$

$$V = \max(R,G,B)$$

③ Count the pixel numbers n_h that have the same Hue value in the sample image. For 8-bit hues, the hue value range is between 0 and 255.

④ Calculate the probability distribution of each Hue value: $P_h = n_h / N$, and get the model of the line color. Where, N is the total of the pixels in the sample image.

Figure 1 shows line color model.
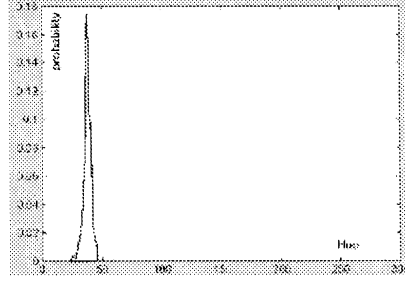
Fig1: Line color model

In the line color model, we choose a probability as threshold. The threshold is decided when detected region of the line is as big as its original region in the raw image. We think that those hues which corresponding probability is bigger than the threshold denote line color, and these hues are named the line hue range. In our case, the line hue range is a const.

When we detect the line region, we first convert the RGB image to the HSV image, and then, in the HSV image, the luminance of those pixels that its hue is in the line hue range are converted to high, and other pixel's probability are converted to low, so that we get the binary image about the line. Figure 2 shows a video image and its binary image about.
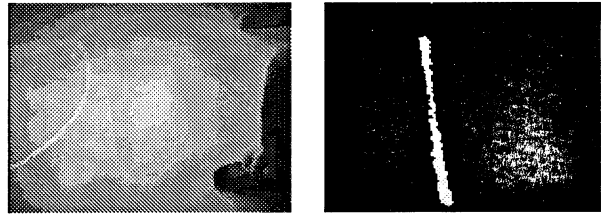
Fig2: A video image and its binary image about line

3 Line tracking

In order to track line, we first must compute the COG (center of gravity) of the line color distribution

of the binary image [1]. The COG of the binary image is found as follows:

Find the zeroth moment

$$M_{00} = \sum_x \sum_y I(x,y)$$

Find the first moment fir x and y

$$M_{10} = \sum_x \sum_y xI(x,y)$$

$$M_{01} = \sum_x \sum_y yI(x,y)$$

Then the COG of the binary image is:

$$x_m = \frac{M_{10}}{M_{00}} \quad ; \quad y_m = \frac{M_{01}}{M_{00}}$$

Where $I(x,y)$ is the pixel (probability) value at position (x,y) in the image, in our case, it's 1.0 or 0, and x and y range over the image. Fig3 shows the COG of the binary image.
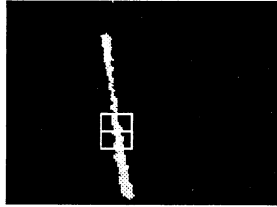
Fig3: The COG of the binary image

The process of line tracking is continuously moving the COG of the binary image to the geometric center of the video frame. But then, for the line tracking of the mobile robot, we use that the robot moving toward the COG of the binary image instead of the moving of the COG. Fig4 shows the principle of the line tracking. l denotes the middle line of the video frame; O is the COG of the binary image; B is robot's situation. The robot is always moving toward the place that corresponding to the COG of the line color distributions in each video scene to accomplish the

purpose that track line for a mobile robot.
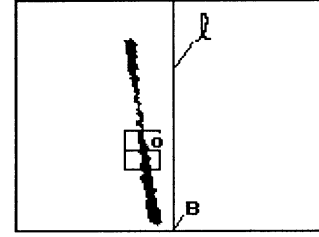
Fig4:   The principle of the line tracking

In our case, robot has two independently driven wheels, and two DC motors independently drives two driven wheel. Fig5 shows the principle of controller
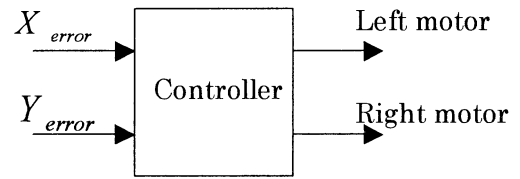
Fig5:   The principle of controller

Where X_{error} and Y_{error} are the offsets between the centre of gravity O and robot B.

$$X_{error} = X_O - X_B$$

$$Y_{error} = Y_O - Y_B$$

The control rule of the controller are shown as follows:

$$f_L = f_R = k_f * Y_{error}$$

$$V_L = f_L + k_t * X_{error}$$
$$V_R = f_R - k_t * X_{error}$$

Where f_L and f_R are the basic speed of left and right wheel when robot moving on straight line; V_L and V_R are the speed of left and right wheel in fact; k_f and k_t are the coefficient of the

controller.

4. Experiment result

In our case, we let the robot to track line on different lighting conditions. The result is almost invariable. Fig6 shows the result of the experiment:
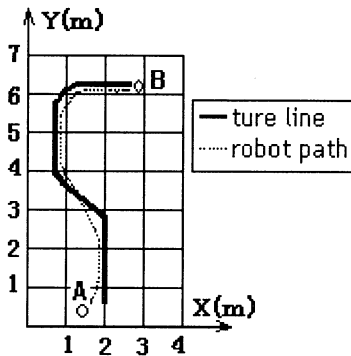
Fig6: The result of the experiment

Where point A is starting point, and point B is terminal.

According to the experiment, we may obtain the result as follow:

1) The effect of line tracking was very good under difference lighting conditions.

2) The effect of line tracking in curved line was worse than straight line.

3) The effect of line tracking will be reduced when there are some non-line objects that have the same color as line in the image.

5. Conclusion

In this paper, a color-based line tracking method was explored. As the line color model was built only by the hue value in HSV color space, so they could reduce the influence of brightness under difference lighting conditions. The results showed this line tracking method were reliable.

But, based on line color model, detected images can only be separated line color regions from non-line color regions. Sometimes, there are some non-line objects that have the same color as line in an image. It will reduce the stability of the line detection and track. In addition, the effect of line tracking in curved line need to improve. These are the next research.

As a result, this method can be applied to other image preprocessing. For example, face detection and track, and landmark detection etc.

References

[1] Gary R.Bradski, "Computer Vision Face Tracking For Use in a Perceptual User Interface," Intel Technology Journal Q2'98.

[2] Jiwu Wang, Masanori Sugisaka, "Study on a color based target detection technique" AROB 7th '02, vol2, pp532-535, Jan, 2002

[3] Masanori SUGISAKA, Renping CHEN, "Map-based the Mobile Robot Navigation in Indoor Environment",2002 FIFA Robot World Congress p.360-363

[4] K. Sobottka and I. Pitas, "Segmentation and tracking of faces in color images," Proc. Of the Second Intl. Conf. On Auto. Face and Gesture Recognition, pp. 236-241, 1996

Gradient Runs Based Guideline Detection Technique

for the Vision System of An Alife Mobile Robot

Jiwu Wang[1]

Department of Electrical and Electronics Engineering,

Oita University, Oita 870-1192, Japan[1]

(Tel : 81-97-554-7831; Fax : 81-97-554-7841)

(Email: wang_jiwu@hotmail.com)

Hidenori Kimura[2,3]   Masanori Sugisaka[1,2]

The Institute of Physical and Chemical Research (RIKEN),

Bio-Mimetric Control Research Center, Shimoshidami,

Moriyama-ku Nagoya, 463-0003, Japan[2]

Complex Systems Department of Complexity Science and

Engineering Graduate School of Frontier Science, The

University of Tokyo, Tokyo 113-8656, Japan[3]

(Email:kimura@crux.t.u-tokyo.ac.jp, msugi@cc.oita-u.ac.jp)

Abstract

A robust and reliable guideline detection system is one prime component for a reliable navigation of a mobile robot with a vision system to determine its position in a map. Color based target detection technique is a simple, rapid and easily realized method. But its satisfactory results can only be achieved under better lighting conditions. And its reliability can be easily influenced by the glisten or poor lighting. In order to reduce the effect of the luminance, color models of the guideline were built in chromatic and HSV color space. The reliability can be improved but was not robust enough, especially under poor lighting conditions. Here gradient runs' detection technique was developed to detect the guideline which color is strongly contrast to the floor. With this technique, the guideline can be detected robustly even under poor lighting conditions. At last, this method was applied in the navigation experiments, and its effect was verified by the experiment results under various lighting conditions.

1. Introduction

Navigation, as a basic and necessary function of a mobile robot, has to often face the problems such as the skid and drift errors of sensors. These make a reliable and smoothly navigation more difficult. The visual based navigation, together with different perceptual landmarks, is an effective solution. This requires the robot should detect, extract and recognize different landmarks accurately under different lighting conditions, especially when there is glisten or poor lighting [1]. In this paper, the visual based navigation method was applied to our Alife robot, which has two CCD video cameras for collecting necessary image information in the surrounding environment. There are two types of artificial landmarks used in the navigation experiments: one is the continuous landmark (guideline), which

is put on the floor to control the robot's trace; others are common landmarks, such as triangle, circle and cross etc, which are used to command the robot to perform some specific movements. Many techniques were applied to the detection and extraction of the landmarks [3]. Color based target detection technique and gradient runs based guideline detection technique were verified to be more robust, reliable and easily realized.

For the color based target detection method, one is based on the threshold values of the target object in the HSV (Hue, Saturation and Value) or chromatic color space. As we known, in the RGB color space, the triple components (R, G, B) represent not only color but also luminance. The luminance is easily affected by lighting conditions. Chromatic colors, also known as pure colors in the absence of luminance, are defined by a normalization of the triple components (R, G, B). Similarly, The Hue Saturation Value (HSV) can also separates out the hue component from components of the saturation and brightness. Another method is to build the color models of some specific targets. Using the color model, we can easily to separate the target color regions from non-target color regions in each captured image. As for the target color regions representing target or not, it should be determined by the recognition processor. The recognition process with these two methods was illustrated in reference [2,4].

Although the luminance can be separated in the HSV color space and chromatic color space, the target objects can not be recognized robustly under the poor lighting or strong glisten conditions. This is because the color of the guideline will unavoidably be influenced by the colors of surrounding objects. Moreover, no matter using threshold color values or color model, they can only separate the target color regions from non-target color regions in each captured image. When there are many objects with the same color of the guideline in some captured images, or there is not enough extracted geometric information of the guideline, the recognition process will be difficult or the guideline can not be recognized.

Here gradient runs technique was discussed in this paper. Because it depends on the intensity change, unlike the guideline color detection methods, it is more effective to detect and extract the guideline under various lighting conditions.

2.Gradient's run detection technique

The gradient method has gotten more applications in the image processing. Here we mainly discuss how to apply it to the guideline detection and extraction. As we know, the gradient change will happen in any part of an image, even for the different parts of the same object in that image. Most of important, the gradient generally will change for different objects in an image. This gives the cues to separate different objects in an image.

In order to realize robust and reliable guideline detection, two parallel guidelines with color highly contrast to the floor, were put on the floor. Because the glistens and poor lighting can not be avoided when the robot moves, and those factors will make effect on the intensity change, the intensity equalization technique was used before the gradient runs' detection to weaken their influence.

$$V_{New} = \frac{V_{Max}}{A_0} \sum_{i=0}^{V_a} H_i$$

where, A_0 is the area of the gray histogram of the original image, H_i is the amount of the ith gray histogram, V_{new} is the new gray scale corresponding the original ith gray level, V_{max} is the highest gray scale in the new image and V_a is the original highest gray scale to be transformed.

Because of the guideline with a color contrasted to the floor color, it is not difficult to detect the edge of the guideline based on the gradient runs technique. Here the gradient runs were produced by the gradient difference between any two pixels in each row. In order to detect the guideline effectively under various lighting conditions, it is necessary to carry on the equalization before gradient runs' detection, especially in the case of glisten or poor lighting. This is because the gradient in such areas of an image is difficult to be detected. After the edge of the guideline was extracted, the guideline will be formed based on the geometric features. The whole image

processing was illustrated in figure 1. After one image was captured, it was transformed into a gray scale image first, equalization, gradient run's detection, and then formed the guideline area. This process was applied in the practical navigation for the visual servo control of our Alife mobile robot.

a) The original image

b) The gray image transformed from the RGB image

c) The gray image after equalization

d) The result after gradient runs' detection
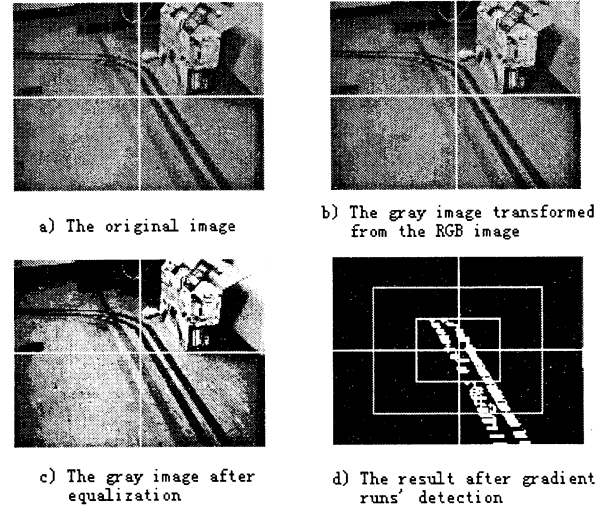
Figure1 An illustration of guideline extraction with gradient's run method

At the same time, we also compare the extraction results of the gradient's run method and color model method. As shown in figure 2, we put the red and blue color guideline on the floor. Figure 1 b) is the extraction result with gradient's run method. We find it can extract the red and blue guideline together effectively. Figure 1 c) is the extraction result with the color model method. It can only separate the red color guideline. This means the guideline will be extracted if there is color gradient contrasted to the floor. Unlike color model, it can only separate the target color regions from non-target color regions.

a) The original image

b) The guideline extraction
with gradient run method

c) The guideline extraction
with color model method
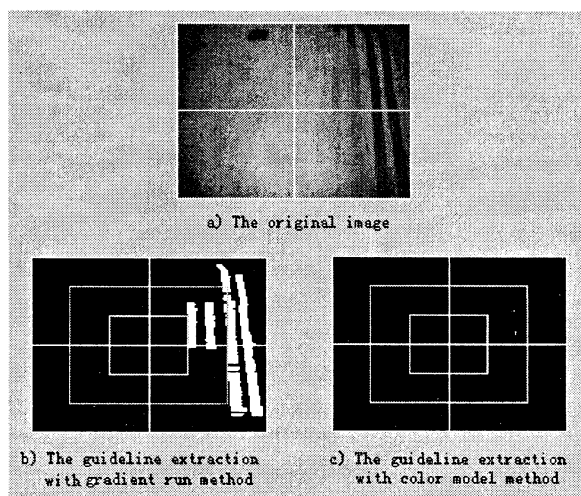
Figure 2 The comparison of the guideline extraction
with gradient's run method and color model method

Figure 3 is an illustration of the landmarks extraction with the gradient's run and color model method respectively. From the results, it can only extract the guideline with the gradient's method, and it can separate target color regions effectively with the color model method. The result also shows how to apply these methods effectively.

a) The original image

b) The guideline extraction
with gradient run method

c) The guideline extraction
with color model method
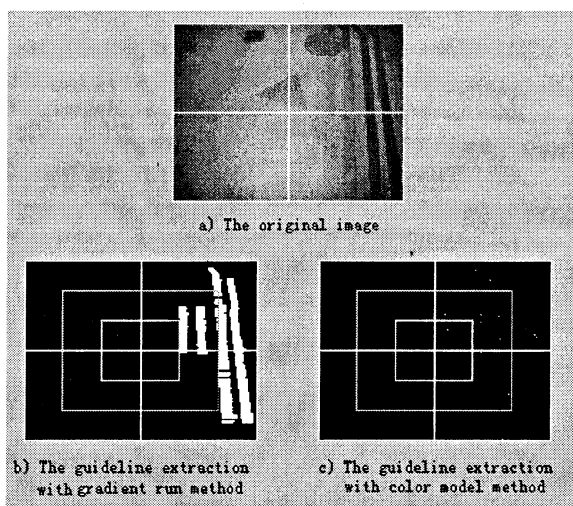
Figure 3 The comparison of landmarks extraction
with gradient's run method and color model method

2. Conclusions

A robust and reliable guideline detection technique is necessary for our Alife robot's navigation. Compared with the color based target detection method, it was verified that the better detection experiment results can be achieved for the gradient runs's technique even under poor lighting and glistening conditions.

Reference

[1] Masanori Sugisaka, Xin Wang, Ju-Jung Lee, Intelligent control with new image processing strategy for a mobile vehicle, Artificial life robotics, 1998: 113-118.

[2] J. Wang, M. Sugisaka, Study on a color based target detection technique", Proceedings of 7-th international symposium in artificial life and robotics (AROB 7-th'02), B-Con Plaza, Beppu, Oita, Japan, Jan. 16-18 : 532-536

[3] J. Wang, M. Sugisaka, Study on visual-based indoor navigation for an alife mobile robot, ICSEng2002, Las Vegas, USA, 2002, August 6-8 : 390-396.

[4] J. Wang, M. Sugisaka, Improvement on the image processing for an automous mobile robot with an intelligent control system, International conference on control, automation and system (ICCAS2001), 2001: 437-440

Chemical Genetic Algorithms - Evolutionary Optimization of Code Translation

Hideaki Suzuki
ATR Human Information Science Labs.
hsuzuki@atr.co.jp

Hidefumi Sawai
Communications Research Laboratory
sawai@crl.go.jp

Abstract

A chemical genetic algorithm (CGA) in which several types of molecules react with each other in a cell is proposed. A cell includes a binary string (DNA) and smaller molecules, and the fundamental mapping from binary substrings on DNA (genotype) to real values for the output parameters (phenotype) is specified by a set of molecules named *aminoacyl-tRNAs*. Through evolutionary modification of genetic information on DNA, the codes on DNA and the genotype-to-phenotype translation coevolve, which enables the optimization of the code traslation during evolution. The CGA is applied to a class of deceptive problems, and its effectiveness is demonstrated in comparison with a simple genetic algorithm (SGA).

Keywords: *genetic algorithms, coding problem, chemical reaction, translation, coevolution*

1 Introduction

From an aspect of the design of an evolutionary system, the translation from genotype to phenotype plays a critical role. The translation specifies the system's genotype-to-phenotype mapping, the fitness landscape on the genotype space, and finally determines the performance of evolution. Although in most studies of artificial evolutionary systems, the translation relation is designated by a (human) designer, if we could make the translation optimized by evolution, we might get a more effective genotype-to-phenotype mapping through evolution.

Genetic algorithms (GAs) [5, 3] are one of the most successful algorithms that have been applied to a variety of engineering optimization problems. However, the conventional approaches of GAs are unsatisfactory in that the translation (mapping from bit substrings to parameter values) is specified by the designer and fixed through evolution. A problem of the design of the translation in GAs is called a 'coding' problem and a number of different coding methods have been proposed and tested [2, 3, 4, 7].

When we consult a biological cell, on the other hand, we find that the translation of genetic information on the DNA strands is not fixed in advance but changed through evolution. The fundamental mapping from the DNA codes (codons) to functional units (amino acids) is specified by a set of molecules named aminoacyl-tRNAs created from the translation of codes on DNA [1]. Because both codes for structural proteins and codes for the translation are present in the same DNA strand, the biological system could have evolved the codes and their translation at the same time.

Borrowing this mechanism, here we propose a genetic algorithm (which we refer to as chemical genetic algorithm, CGA) that enables the coevolution between codes on DNA and their translation [6]. We prepare a population of artificial cells which include four different types of molecular units: a DNA string, aminoacyl-tRNA units (aa-tRNAs), tRNA strings, and amino acid units (Aminos). A population of cells having this structure is evolved by using operations for selection, DNA mutation, DNA crossover, molecular exchange, and chemical reaction. The cell's fitness is evaluated from the target function, which is calculated from the specific output amino acid values. To assess the validity of the CGA, we apply it to a functional optimization problem that are hard to solve by a simple GA (SGA). Numerical experiments show great advantage of CGA vs. SGA for the tested optimization problem.

The organization of the paper is as follows. After explaining a proposed model for the CGA in detail in Section 2, Section 3 introduces a deceptive function and describes experimental conditions and results. Some concluding remarks and discussions are given in Section 4.

2 The Model

CGA is a method to evolve a population of cells which have a structure shown in Fig. 1. The entire procedure of the algorithm is as follows:
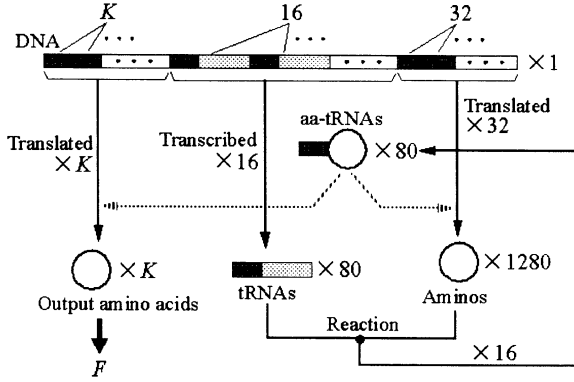
Figure 1: A cellular structure used in the CGA. Dark hatched rectangles are codons described with a short binary strings, bright hatched rectangles are indexes described with binary strings, and circles are amino acids described with real numbers. The total length of a DNA string is $KJ_1 + 16(J_1 + J_2) + 32J_1$, where J_1 is the codon length and J_2 is the index length.

1. **[Initilization]** Prepare a population of cells with the archtecture shown in Fig. 1. In the initial state, each cell only includes a DNA string and Aminos which are created randomly.

2. **[Chemical Reaction]** Create 16 tRNAs and 32 Aminos by the transcription/translation of a DNA string. The newly created molecules are mixed with older molecules, and then tRNAs and Aminos react with each other. A pair of randomly chosen tRNA and Amino is compared and, if the tRNA's index matches the Amino's value based on a predefined criterion, a new aa-tRNA is created. This process is repeated several times, and 16 newly created aa-tRNAs are mixed with older aa-tRNAs. After these processes, some of the smaller molecules are randomly chosen and eliminated to limit the numbers of the smaller molecules in a cell to constant numbers.

3. **[Selection]** Calculate the fitness value for each cell using the translation specified by a set of aa-tRNAs. For each codon in the DNA string, an aa-tRNA with the same codon is chosen from the aa-tRNA set, and its amino acid value is used to calculate the output function. Using these function values, roulette-wheel selection is conducted among the cells.

4. **[DNA Mutation]** Conduct the conventional mutation (bit flipping) operation on the DNA strings

of the cells.

5. **[DNA Crossover & Molecular Exchange]** Mate a particular portion of the cells and conduct the conventional DNA crossover operation and a molecular exchange operation (50

6. Terminate if a particular condition is satisfied. Otherwise, go to Step 2.

At the outset, every cell has 1280 different (but common for all cells) Aminos. This number is large enough to make the amino acid diversity stored in the Aminos sufficiently large. As evolution goes on, the diversity gradually decreases, and at the same time, the amino acid diversity in the aa-tRNAs increases. If by chance an appropriate aa-tRNAs is created by the reaction in a cell, the cell gets a higher fitness value than the others and its genetic information on the DNA strand and smaller molecules begins to spread not just through reproduction in the selection operation but also through the molecular exchange operations. An appropriate set of amino values is chosen in this way from a large initial amino acid repertory.

3 Experiments

To verify the effectiveness of the CGA, we solve a functional optimization problem using the CGA and the SGA. The problem is a K-dimensional deceptive function which is difficult to solve by conventional GAs such as the SGA. The fitness function is defined as $fitness = a + bF(x)$ for linear scaling or $fitness = \exp(cF(x))$ for exponential scaling where a, b and c are constant values. $F(x)$ is defined as:

$$F(x) = \left(\frac{1}{K} \sum_{k=0}^{K-1} f_k(x) \right)^5,$$

where $f_k(x)$s are functions shown in Fig. 3.

A DNA string in the SGA is the left part of the CGA string. Its length is $J_1 \times K$ bits, and the codons in the SGA are interpreted as real values by using the binary coding method. p_m (mutation rate) and p_c (crossover rate) were respectively set to 0.005 and 0.7 in common for both GAs, and the number of cells (i.e., population size) was set to 256. Ten different runs with different random seeds were conducted for the SGA and CGA in both five- and ten-dimensional cases.

Table 1 compares results between the SGA and CGA. The case of $J_1 = 10$ in the SGA approximately
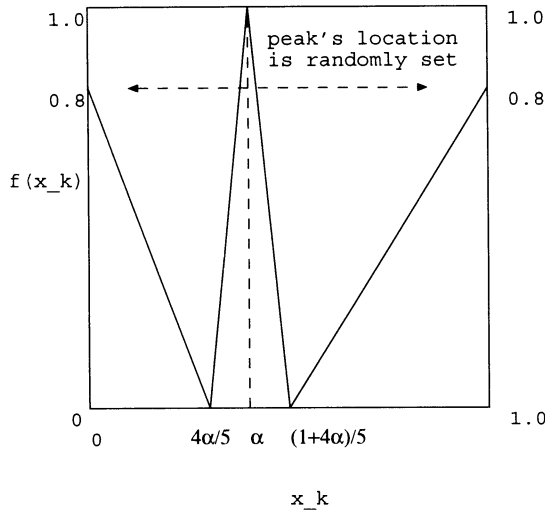
Figure 2: $f(x_k)$ for a deceptive function. α_k is a different random number between 0 and 1 depending on each dimension k $(k = 0, 1, \cdots, K - 1)$. The attractor with a global optimum only has a length of 0.2, whereas the attractors with local optima have much more and wider regions in K-dimensional space.

corresponds to the case of the CGA with initial 1280 different amino-values. Fitness was defined as linear scaling with $a = 0$, $b = 1$ for the SGA and as exponential scaling with $c = 20.79$ for the CGA. Computational cost for the SGA and CGA to run 1000 generations is respectively a few minutes and 20-30 minutes using a general-purpose SUN workstation. Therefore, the computational cost of the CGA is about ten times larger than that of the SGA. Comparing the results between the SGA and CGA, almost all runs of the CGA were successful for the five-dimensional deceptive problem and a half of runs of the CGA were successful for the ten-dimensional deceptive problem, whereas the SGA never succeeded aside from one run.

Figures 3 and 4 show the function values $F(x)$ for the runs of the CGA and SGA, respectively. For the CGA, diversity during evolution of the CGA is well maintained because the best and average values are different from each other. Although the function values of the CGA rose up slowly, both the best and average values became better and better and finally reached a global optimum. For the SGA, on the other hand, almost all runs converged to local optima without converging to a global optimum because diversity in the function values is small due to the fact that the average and best values are approximately the same.

Table 1: Success ratio in SGA and CGA

Algorithm	SGA	CGA	SGA	CGA
Dimension (K)	5	5	10	10
Codon length (J_1)	10	4	10	6
Fitness scaling	linear	exp.	linear	exp.
Sccess ratio	1/10 (10%)	10/10 (100%)	0/10 (0%)	5/10 (50%)

For each condition, the number of successful runs out of ten different trials is shown with the success ratio in parentheses. For example, 1/10 (10%) means that one run out of ten runs succeeded in finding the global optimum, so the success ratio is 10%.

4 Conclusion and Discussion

We have developed a new bio-molecular algorithm, a chemical genetic algorithm (CGA), in which several types of molecules react with each other in a cell. Translation from codons in DNA to amino acids is specified by a particular set of translation molecules (aminoacyl-tRNAs), which are created by the reaction between tRNAs and amino acids. Those smaller molecules are created from DNA, so the codes in DNA and the code translation in smaller molecules coevolve in the model. During evolution, the fundamental genotype-to-phenotype mapping is adaptively changed and converges to the optimum one. Through the struggle between cells with a DNA strand and smaller molecules, a specific output function (protein), which is used to evaluate a cell's fitness, is optimized. The effectiveness of the CGA as compared to the SGA was demonstrated using the numerical experiments with a set of deceptive problems.

According to the main results shown in Table 1, we can say that for the tested deceptive functions, the CGA can find a global optimum solution far more often than can the SGA. This great advantage of the CGA over the SGA comes from the appropriate control of diversity in the population. As shown in Fig. 3, the CGA can recurrently generate a variety of amino acid values even after a population is stuck around the vicinity of a local optimum. Since the output amino values are affected by both the DNA sequence and the smaller molecules, the CGA can explore the search space more extensively than the SGA, which enables the CGA to escape from local optimums. This is especially the case for an early stage of evolution. For
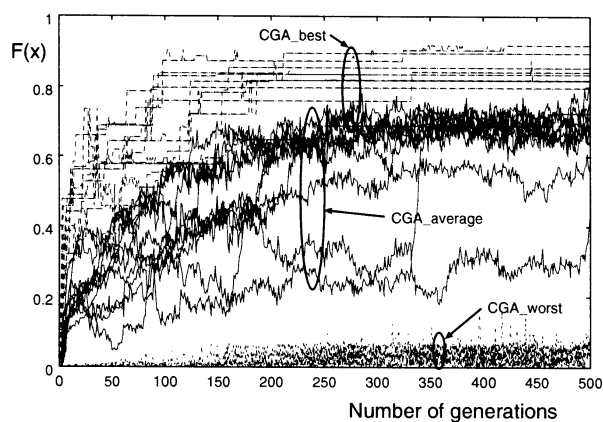
Figure 3: Evolution of CGA for the five-dimensional deceptive problem. $J_1 = 4$. The solid fluctuating lines represent the average values, the doted lines represent the best values, and the lower fluctuating dashed lines represent the worst values. Ten different runs are superimposed.

Figure 4: Evolution of SGA for the five-dimensional deceptive problem. $J_1 = 10$. See the Fig. 3 caption for the description of lines.

later stages of evolution, on the other hand, the CGA exhibits another behavior, the convergence to the optimum repertory of amino acids. As evolution goes on, the diversity in amino values gradually decreases. The coevolution between DNA and smaller molecules controls this convergence in an appropriate manner, so that the CGA can finally obtain the optimum translation relation. For the SGA, on the other hand, once a population is stuck at a local optimum, evolution enters a long period of stasis due to the lack of cell diversity as shown in Fig. 4. Because the translation relation from DNA to real output values is fixed through a run, it is very hard for the SGA to escape from a local optimum wherein a population of DNA strands loses the diversity at some loci. The SGA offers far smaller possibility of exploration than the CGA does. Although the computational cost of the CGA is about ten times larger than that of the SGA, the excellent property of the CGA in controlling diversity makes the CGA a profitable algorithm for difficult problems such as deceptive problems.

References

[1] Alberts, B., Bray, D., Lewis, J., Raff, M., Roberts, K., Watson, J.D.: 1994. *Molecular Biology of the Cell, The Third Edition*. New York: Garland Publishing (1994)

[2] Caruana, R.A., Schaffer, J.D.: 1988. Representation and hidden bias: Gray versus binary coding in genetic algorithms. In: Laird, J. (ed.): *Proceedings of 5th International Conference on Machine Learning*. San Mateo, CA: Morgan Kaufmann (1988) 153-161

[3] Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. New York: Addison-Wesley (1989)

[4] Goldberg, D.E., Korb, B., Deb, K.: Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems* **3** (1989) 493-530

[5] Holland, J.H.: *Adaptation in Natural and Artificial Systems*. Boston: MIT Press (1992)

[6] Suzuki, H., Sawai, H.: Chemical genetic algorithms - Coevolution between codes and code translation. To be published in: Proceedings of The 8th International Conference on the Simulation and Synthesis of Living Systems (Artificial Life VIII) (2002)

[7] Wright, A.H.: Genetic algorithms for real parameter optimization. In: Rawlins, G.J.E. (ed.): *Foundations of Genetic Algorithms (FOGA-1)*. San Mateo, CA: Morgan Kaufmann Publishers (1991) 205-218

Evolution of Cooperation with a Dynamically Separating Mechanism of Individuals

Koichi NAKAYAMA[1],[2] Hideaki SUZUKI[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 propose a dynamically separating mechanism of individuals and describe how the mechanism promotes the evolution of cooperation among individuals. An individual is modeled as an agent that ingests materials, generates energy and performs metabolic functions, donation, and consumption. Using a genetic algorithm incorporating a dynamically separating mechanism, we simulate the evolution of the multi-agent system. The simulations show that the evolution of cooperation is effectively promoted by the dynamically separating mechanism of individuals.

1 Introduction

The evolution of cooperation is one of the most important research issues in the field of artificial life, because the evolution of cooperation should play a crucial role in all kinds of symbiosis, from the cell level to the ecological level. In this paper, we model the evolution of cooperation as an evolutionary process in which independent individual entities in a system cooperate by sharing and/or partitioning some functions and eventually become indispensable to each other's lives.

Cooperation is a kind of relationship or interaction between individuals. Such interactions should be affected by the environment and, strictly speaking, environmental factors must be considered in the modeling of the interactions. In this paper, however, leaving the issue of relationship between interactions and the environment as a future topic of study, we focus on interactions between individuals, especially on the effect of a dynamically separating mechanism of individuals.

A dynamically separating mechanism of individuals is achieved by dynamically separating individuals into subgroups called colonies so that the interactions of the individuals are limited within the colony to which the individuals belong. In addition, colonies are separated or integrated recursively depending on the number of individuals. The Dynamically Separating Genetic Algorithm (DS-GA) is a genetic algorithm that incorporates the dynamically separating mechanism of individuals [1][2].

We then discuss how the dynamically separating mechanism of individuals promotes their evolution of cooperation. Concretely, an individual is modeled as an agent that ingests materials, generates energy and performs metabolic functions, donation, and consumption. Using the DS-GA and conventional GAs without a dynamically separating mechanism, we simulate the evolution of the multi-agent system. By comparing the simulation results with the DS-GA and GAs, it will be seen that the evolution of cooperation is effectively promoted by the dynamically separating mechanism of individuals.

2 Simulation Model

In this section, we model the dynamically separating mechanism as the DS-GA, and the activities of lives that have predominancy by the evolution of cooperation as the 2M-E Model, which is one of the multi-agent systems. We also model a mechanism in which an expression of some kind appears from an individual with the same genotype using a regulator of the genome.

2.1 Dynamically Separating Mechanism

In researching multi-agent systems, the dynamically separating mechanism was modeled as the DS-GA[1][2]. We use the DS-GA as a model incorporating the dynamically separating mechanism. The DS-GA is a model that dynamically separates individuals into subgroups called colonies. The separation state changes dynamically according to the number of individuals. In nature, the number of individuals that each individual can contact is limited. Especially, the following environments may contain a dynamically separating mechanism. A)The air bubbles which rise from an underwater volcano. B)The colony of fungi which will be separated if it grows above some extent.

As an algorithm in comparison with the DS-GA, we use two kinds of typical GA, a simple GA and an island model GA [3] without a dynamically separating mechanism.

2.2 Activities of Lives

In order to investigate the validity of the evolution of cooperation, we model activities of lives as a two-material and energy model (2M-E Model) which has predominancy by the evolution of cooperation. The 2M-E model is a model which simplifies lives by ingesting two materials and generating energy using each function. The model has predominancy by cooperation as in sharing and partitioning some functions.

We determined the number of functions arbitrarily as five. In the 2M-E model, an individual consumes energy E according to the functions which it has in every time step. E is generated from two kinds of materials, A and B, e.g., oxygen and a nutrient, which are ingested from the environment. We assume that sufficient materials are always available. Each individual can have some of the five following functions. (1) The ingesting function of material A from the environment (*Function A*). (2) The ingesting function of material B from the environment (*Function B*). (3) The generating function of energy E from two kinds of materials (*Function E*). (4) The donating function of A, B, and E to neighbors (*Function D*). (5) The reproducing function in which an individual reproduces (*Function R*). The functions that an individual has are decided with a phenotype.

2.2.1 Individuals' Action

At every time step, individuals perform (1) metabolic functions, (2) donation, and (3) consumption. An individual for which energy is insufficient disappears and an individual with a reproducing function in which energy increases reproduces.

The state of the a-th individual at time t is represented by $S_A(a, t)$, $S_B(a, t)$, and $S_E(a, t)$, which denotes the amounts of A, B, and E in it respectively.

(1)Metabolism

At every time step, an a-th individual performs one of the functions $\{A, B, E\}$ chosen at random from the functions that an individual has. The individual with no functions does no metabolic function. When ingesting one of the materials, the amount of remaining materials changes according to formula (1).

● Ingesting one of the materials $\{A, \text{ or } B\}$

$$S_{\{A, or B\}}(a, t) = S_{\{A, or B\}}(a, t - 1) + 100 \quad (1)$$

In generating energy E, the amounts of remaining materials and energy change according to formula (2).

● Generating energy E

$$
\begin{aligned}
S_A(a, t) &= S_A(a, t - 1) - 50 \\
S_B(a, t) &= S_B(a, t - 1) - 50 \\
S_E(a, t) &= S_E(a, t - 1) + 50 \quad (2)
\end{aligned}
$$

If either $S_A(a, t)$ or $S_B(a, t)$ is less than 50, it does nothing.

(2)Donation

At every time step, an a-th individual with a donating function donates half of each material A,B, and energy E to the other individuals. The donated materials and energy are shared by the other individuals in the same colony. The change of the amount of $S_A(a, t)$, $S_B(a, t)$, and $S_E(a, t)$ by donation is shown in formula (3).

N shows the number of individuals in a colony.

$$
\begin{aligned}
S_{\{A,B,E\}}(a, t) &= S_{\{A,B,E\}}(a, t - 1) \times 0.5 \\
S_{\{A,B,E\}}(i, t) &= S_{\{A,B,E\}}(i, t - 1) \\
&\quad + \frac{S_{\{A,B,E\}}(i, t - 1) \times 0.5}{N - 1} \\
&\qquad (i \in \{1, 2, \cdots, N\}, i \neq a) \quad (3)
\end{aligned}
$$

(3)Consumption

At every time step, an a-th individual consumes energy E according to the function that the individual has. If it has *Functions A, B, and E*, it consumes two units of energy for each function respectively. And if it has *Function R*, it consumes ten units of energy.

The change of the amount of energy S_E by consumption is shown in formula (4).

$$
S_E(a, t) = S_E(a, t - 1) \\
- \sum_{\{A,B,E,D,R\}} \left\{ \begin{array}{l} 0(FunctionD) \\ 2(FunctionA, B, E) \\ 10(FunctionR) \end{array} \right.
$$

$$(4)$$

2.2.2 Reproduction and Extinction

The a-th individual reproduces or becomes extinct depending on the amount of Energy $S_E(a, t)$. An individual will be extinguished when $S_E(a, t)$ becomes zero or less. An individual with the *function R* will reproduce an individual with the same genotype as the original individual's genotype, if $S_E(a, t)$ becomes more than twice as much as the initial amount $S_E(a, 0) = 50$. The two individuals inherit half of the original

Table 1: Five bits of phenotype in 2M-E Model.

	bit 1st	bit 2nd	bit 3rd	bit 4th	bit 5th
Function	ingesting A	ingesting B	synthesizing E	donation A, B, E	reproduction
ability gene	1(Yes) or 0	1(Yes) or 0	1(Yes) or 0	1(Yes) or 0	1(Yes) or 0

individual's amount of $S_E(a,t)$, $S_A(a,t)$ and $S_B(a,t)$. At that time, genes are mutated according to the mutation probability P_{mut} for every bit of genes.

2.3 Regulator of the Genome

A genotype and a phenotype exist in actual living genes. In a certain development process in actual life, the expression pattern of the gene changes due to the influence of a nearby cell[4]. Based on regulatory genes, the expression of those genes is changed into a phenotype from a genotype. As a result of these influences, the expression pattern of genes is mutually controlled. Especially for the evolution of cooperation, a mechanism that mutually controls the expression of genes is important. In this paper, the following regulator techniques are used.

2.3.1 Phenotype

In this model, factors which determine the individuals' capability are the existence of five functions (*Functions A, B, E, D, and R*) . We use these five bits as a phenotype. The phenotype of each individual is represented in Table 1. A "1" for each bit represents the existence of the function.

2.3.2 Genotype

⟨Composition of the Genome and Regulator⟩

The genome consists of the regulatory genes' part and the structural genes' part. Moreover, three kinds of repressor protein, $r_\alpha, r_\beta, r_\gamma$, and enhancer protein, $e_\alpha, e_\beta, e_\gamma$, are used as regulatory proteins. In this experiment, the mechanism that mutually controls the expression of genes is modeled as follows. (1)In order to determine the phenotype, each individual has three sets of structural genes. (2)In order to determine the expression of three sets of structural genes, each individual has eight sets of regulatory genes. (3)In order to determine the expression of a set of regulatory genes, each individual takes the regulatory proteins of another individual chosen at random from the same colony. As a result, an individuals' regulatory proteins choose another individuals' set of regulatory

genes. The set of regulatory genes synthesize regulatory proteins, and choose a set of structural genes. And the set of structural genes determines the phenotype. By repeating this mechanism, the expression pattern of the genes is controlled mutually. See the following for details.

⟨ Structural Genes' Part ⟩

In the structural genes' part, there are three promoters, SP_α, SP_β, and SP_γ, and three sets of structural genes, SG_α, SG_β, and SG_γ. Each set of structural genes is expressed by each promoter when the enhancer protein is synthesized in the regulatory genes' part. For example, when promoter SP_β is expressed, the set of regulatory genes SG_β is expressed. In this model, some of the three promoters are expressed, and then some of the three sets of structural genes are expressed.

The three promoters are expressed by three kinds of enhancer protein e_α, e_β, and e_γ, and each set of structural genes consists of five bits, 1, 2, 3, 4, and 5. For example, the set of structural genes SG_α has five bits, $SG_{\alpha 1}$, $SG_{\alpha 2}$, $SG_{\alpha 3}$, $SG_{\alpha 4}$, and $SG_{\alpha 5}$. There are 15 bits of structural genes in total (Fig. 1).

The value of the structural genes shows the existence of a function. A "1" for the structural genes shows the existence of a function, and a "0" for the structural genes shows the non-existence of a function. In this experiment, "1" is made dominant and "0" is made recessive. It will be referred to as 1 if at least one is 1 when the two or more sets of structural genes are expressed (OR processing). For example, if two sets of structural genes "10101" and "11100" are expressed, the phenotype is "11101".

⟨Regulatory Genes' Part⟩

In the regulatory genes' part, there are eight promoters, $RP_1, RP_2, \cdots, RP_8$, and eight sets of regulatory genes, $RG_1, RG_2, \cdots, RG_8$. Each set of regulatory genes is expressed by each promoter. For example, when promoter RP_7 is expressed, the set of regulatory genes RG_7 is expressed. In this model, one of the eight promoters is expressed, and then one of the eight sets of regulatory genes is expressed.

Each promoter consists of three bits, α, β, and γ, and each set of regulatory genes consists of three bits, α, β, and γ. For example, the set of regulatory genes with the number RG_7 has three bits, $RG_{7\alpha}$, $RG_{7\beta}$, and $RG_{7\gamma}$. There are 24 bits of regulatory genes in total (Fig. 1).

The value of the promoters' bit shows the expression condition. A "1" for $RP_{n\alpha}$ shows the enhancer protein e_α and a "0" for $RP_{n\alpha}$ shows the repressor protein r_α. The value of the regulatory genes shows
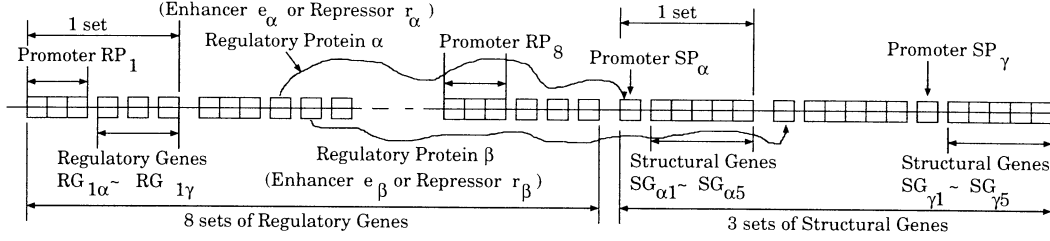
Figure 1: Conceptual representation of the Genome.
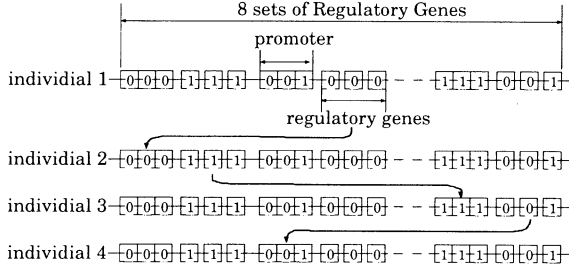
Figure 2: Regulatory Relation between Genome

the type of synthesizing regulatory protein. A "1" for $RG_{n\alpha}$ shows the synthesizing of enhancer protein e_α and a "0" for $RG_{n\alpha}$ shows the synthesizing of repressor protein r_α. When three kinds of regulatory protein satisfy the condition of the promoters' three bits, the promoter is expressed, and the promoter expresses a set of regulatory genes. The expressed set of regulatory genes synthesizes regulatory protein according to the value of the regulatory genes. For example, if an individual chosen at random from the same colony has three kinds of proteins r_α, e_β, and r_γ, the promoter which has "010" is expressed. And, if the expressed set of regulatory genes is "101", the sets of regulatory genes synthesize the three proteins e_α, r_β, and e_γ. In this model, eight sets of promoters correspond to all of the patterns $2^3 = 8$ of the three kinds of regulatory protein. In addition, only the regulatory genes and the structural genes mutate. Promoters do not mutate.

2.4 Implementation

The implemented model is as follows. $100,000$ individuals at $t = 0$ are created and separated into colonies. The number of individuals in a colony is 10. The accumulated profit of an a-th individual, $S_A(a,t)$,$S_B(a,t)$,$S_E(a,t)$, is initially set to 50 and its genes are initially randomly chosen. The mutation probability is set to $P_{mut} = 0.01$, the migration prob-

ability is set to $P_{mig} = 0.001$ for all individuals. When the total number of individuals that can exist in this model becomes greater than the initial number of individuals, a colony is eliminated at random.

3 Experiments

In this experiment of DS-GA, three kinds of phenotype dominate the cooperation states all 20 times. Among those 20 times, the following three kinds of phenotype dominate 14 times. In this experiment of a simple GA and an island model GA without a dynamically separating mechanism, one kind of phenotype [1,1,1,0,1] dominates all 20 times.

As an example of the result of DS-GA, figure 3 shows the number ratio of phenotypes.
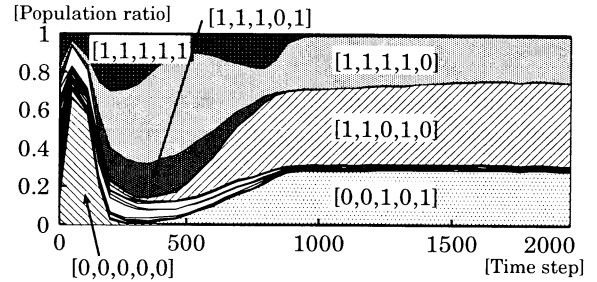
Figure 3: Example of Population ratio

Three kinds of expression patterns had a large number ratio, and these were stabilized. The phenotypes which had a large number ratio are shown below.

[1,1,1,1,0]All-Assistant : An individual of this phenotype can ingest **A** and **B** and generate **E**. The materials and energy are donated to other individuals. It does not reproduce. Consequently, this individual continues assisting other individuals.

[1,1,0,1,0] AB-Assistant : An individual of this phenotype can ingest **A** and **B**. **E** is not generated. The

materials and energy are donated to other individuals. It does not reproduce. Consequently, this individual continues assisting other individuals with **A** and **B**. **[0,0,1,0,1] E-Generating and Reproducer** : An individual of this phenotype generates **E** and reproduces. **A** and **B** are not ingested. The materials and energy are not donated to other individuals. Consequently, this individual continues reproducing, having other individuals assisted with **A**, **B**, and **E**.

When these three kinds of individuals exist independently, neither can survive. The All-Assistance Individual and AB-Assistance Individual acquire reproducing capability with E-Generating and Reproducting Individuals. The genotype at this time which had the largest number is [0,0,1,1,0], [1,1,0,1,0], [0,0,1,0,1]. The phenotype interaction by the promoter of the regulatory genes' part at this time is shown in Figure 4.

Promoter	Phenotype
[0,0,1]	[0,0,1,0,1]
[0,1,0]	[1,1,0,1,0]
[1,0,0]	[0,0,1,1,0]
[1,1,0]	[1,1,1,1,0]
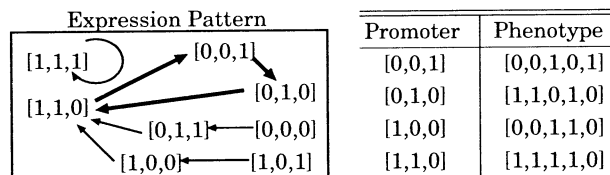
Figure 4: Intaraction among Regulatory Genes

Three kinds of expression patterns, [0,0,1],[0,1,0], and [1,1,0], circulated. Expression patterns other than [1,1,1] induce [1,1,0]. Consequently, it evolved so that three kinds of expression patterns mostly appear. And, it turns out that three kinds of phenotypes appear almost equally from the same genotype.

The experiment results applying the DS-GA show a cooperative state between individuals. However, the experiment results applying a simple GA and island model GA without a dynamically separating mechanism do not show a cooperative state between individuals. From these results, the dynamically separating mechanism is considered to have promoted the evolution of cooperation.

4   Conclusion

In this paper, we proposed a dynamically separating mechanism of individuals and investigated how the mechanism influences the evolution of cooperation of individuals. That is, a dynamically separating mechanism of individuals was implemented by dynamically separating individuals into subgroups called colonies and by making colonies separate or integrate recur-

sively depending on the number of individuals. We also conducted simulations on the evolution of cooperation using the Dynamically Separating Genetic Algorithm (DS-GA) incorporating the dynamically separating mechanism of individuals and the traditional GAs for comparison.

Through the simulations we could verify that cooperative relationships between individuals appear when the dynamically separating mechanism is applied, while no cooperative relationship appears without the dynamically separating mechanism. Thus, we can conclude that the evolution of cooperation is effectively promoted by the dynamically separating mechanism of individuals.

As mentioned in the introduction, the most effective and reasonable factor for separating individuals into subgroups like colonies should be grounded on the environment. Moreover, the environment should have an influence on the interactions between individuals as well as the activity of their lives. The next step of this study, therefore, will model interactions between individuals and the environment.

Acknowledgements

References

[1] K. Nakayama, K. Shimohara, O. Katai, "Dynamically Separating GA: A New Method of Gaining the Global Optimal Solution in MAS," *Proceedings of the Fifth International Conference on Human and Computer*, pp. 204-209, 2002.

[2] K. Nakayama, H. Matsui, Y. Nomura, "Dynamically Separating GA," *IPSJ Transactions on Mathematical Modeling and Its Applications* , Vol. 43, No. SIG 10 (TOM 7), pp. 95-109, 2002.

[3] Cohoon, J. P., Martin, W. N., Richards, D. S., "A Multi-population Genetic Algorithm for Solving the K-Partition Problem on Hyper-cubes," *Proc. 4th ICGA*, pp. 244-248, 1991.

[4] Spemann, H., Mangold, H. "Induction of embryonic primordia by implantation of organizers from a different species" *Roux's Archiv*, Vol. 100, pp. 599-638, 1924.

An Index of Degrees of Confusion between Knowledge Acquired in a Learning Classifier System

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

* Kyoto University Graduate School of Informatics Yoshida Honmachi, Sakyo-ku, Kyoto 606-8501 Japan

† ATR Human Information Science Laboratories 2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288 Japan

‡ Tokyo Institute of Technology 4259 Nagatsuta-cho, Midori-ku, Kanagawa 226-8502 Japan

E-mail: hirinoue@his.atr.co.jp

Abstract

This paper proposes an index of degrees of confusion between knowledge that is acquired by two agents implemented by Learning Classifier Systems. This index can signify degrees of confusion for knowledge which has generality, and it can be used when agents select the best partner for knowledge sharing in a multi-agent system. We showed that this index has great potential in estimating degrees of confusion through experiments where agents are implemented by Learning Classifier Systems and environments are described by finite state machines.

1 Introduction

How can we improve the performance of multi-agent learning systems[5]? There are two ideas that can improve the performance of multi-agent learning systems. The first is that agents exchange acquired knowledge with each other. However, a partner exchanging knowledge should be carefully chosen because the knowledge an agent accepts may conflict with the knowledge that the agent originally has. The second idea prevents declining performance, where agents substitute a malfunctioning agent. If no agent substitutes the malfunctioning agent, the knowledge acquired by the agent that is malfunctioning will be discarded. Thus, a system with substituting agents has an advantage compared to other systems in terms of performance. Just as exchanging agents are chosen, a substituting agent should also be carefully chosen, so that the received knowledge does not contradict the original knowledge. Nevertheless, it is an open issue whether agents can exchange or receive knowledge because designers cannot know beforehand what knowledge agents have acquired.

There is research on multi-agent learning systems. For example, research on Organizational-learning oriented Classifier System (OCS)[3] and cooperative agents[4] discussed sharing knowledge in multi-agent learning systems. However, there has been little research dealing with the effective selection of agents exchanging or receiving knowledge in multi-agent learning systems.

One approach to selecting agents that exchange and receive knowledge is measuring the degree of confusion between the knowledge the two agents have. The degree of confusion means that the number of rules which indicate different actions against the same sensor states (knowledge is signified by condition-action rules in multi-agent learning systems.). For example, when exchanging knowledge, the agent calculates the degrees of confusion between itself and all the other agents; it can then select the agent whose knowledge has the least degree of confusion. This way of using degrees of confusion is also applicable to receiving knowledge.

We propose an index for degrees of confusion between knowledge that is acquired by two reinforcement learning agents.

This paper is organized as follows. Section 2 starts by describing the measurement of degrees of confusion. Section 3 describes the agent architecture and the environment. Section 4 presents the experiments and Section 5 discusses the effectiveness of our index. Section 6 concludes the paper.

2 Measuring Degrees of Confusion

2.1 Degrees of confusion in multi-agent learning systems

Knowledge on reinforcement learning is based on condition-action rules and their values[1]. Thus, exchanging and receiving knowledge can be achieved by merging the knowledge two agents have. Merging

---

[1] In this paper, 'reinforcement learning' can be exchanged with 'Learning Classifier System without the Genetic Algorithm mechanism'. The Learning Classifier System is described in Section 3.

knowledge corresponds a state where (1) condition-action rules are merged and one where (2) the values of rules for each piece of knowledge are summed up. Confusion in merged knowledge is defined as a state where there are two rules whose conditions are identical but the actions are different. When an agent has a great deal of confusion and acts in the environments where knowledge has been acquired, the performance of the agent is probably lower than it was before it merged.

In addition, there is an essential function for rules in multi-agent learning systems. This is general rules. In multi-agent learning systems, there is a problem known as the explosion of state space. This is because other agents' actions should be taken into account. Thus, the calculation of learning increases exponentially. Using general rules can decrease this problem, since they can omit the condition's elements and reduce the calculations for learning. Figure 1 has examples of general and non-general rules. The # represents *don't care*, which is used to ignore the element of a condition, i.e. # is a function enabling general rules to be used. Concretely, the upper left general rule is equal to the non-general rules at the two upper right. This is the meaning of general rules. Confusion between general rules is not as clear as the one between non-general rules, since one general rule corresponds to many non-general rules.
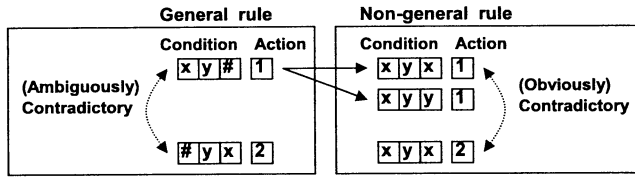
Figure 1: Example of general and non-general rules

2.2 Calculation of index

The ultimate objective is calculating degrees of confusion from the knowledge agents have acquired. However, knowledge in reinforcement learning not only has rules, but also values for these rules. Thus, if there are contradictory rules between rule sets, the degree of confusion depends on those values, and calculating degrees of confusion is not simple.

To cope with the above problem, we propose an index for degrees of confusion between rules of knowledge with general rules. The index estimates the degrees of confusion. This is an estimate based on essential rules extracted from environments. Rules extracted from one environment are equal to rules which

have high values after adequately learning in reinforcement learning. Figure 2 shows an example. Extracted rules approximately correspond to prospective rules acquired by agents, but do not contain general rules. Thus, the effect of general rules should be taken into account in calculating the index.
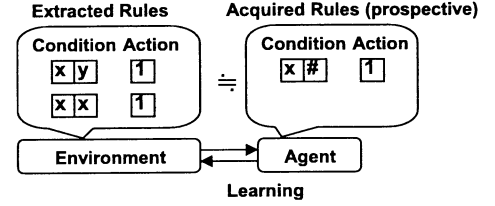
Figure 2: Relationship between extracted rules and acquired rules

The index is calculated as follows:

$$\text{Index} = \sum_{k \in A, l \in B} f(k, l)$$

$$f(k, l) = \begin{cases} 0 & \text{if Act}(k) = \text{Act}(l) \\ \frac{1}{\text{Dist}(k,l)+1} & \text{if Act}(k) \neq \text{Act}(l) \end{cases}$$

where A and B are rule sets extracted from environments, k and l are rules, $\text{Act}(k)$ is an action of rule k, and $\text{Dist}(k,l)$ is the number of different elements in the conditions between rules k and l. Function f signifies the amount of confusion between two rules. If actions of rules are identical, these rules do not cause confusion ($f(k,l) = 0$). In contrast, if actions are not identical, these rules can cause confusion ($f(k,l) = \frac{1}{\text{Dist}(k,l)+1}$). That is, the more similar the conditions of two rules are, the more confusing those rules are, since agents may acquire general rules which ignore different elements of conditions. Figure 3 shows an example whose $\text{Dist}(k,l)$ is 1. An agent may acquire a right-hand rule against left-hand rule k (in extracted rule set A), and this rule is contradictory to rule l (in extracted rule set B) (because actions are different.). The probability, such as in this case, increases if Dist is low. Finally, function f is calculated for all combinations between extracted rule sets A and B, and the index is the sum of these.
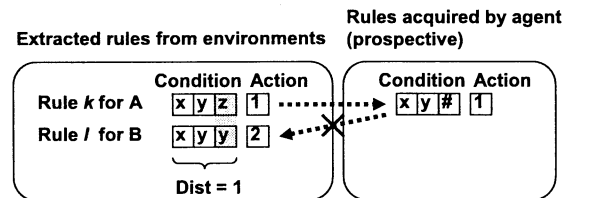
Figure 3: Effect of distance between conditions

3 Agent Architecture and Environment

3.1 Learning Classifier System

A Learning Classifier System (LCS)[1] is a general rule-based learning system in which a set of condition-action rules called *classifiers* compete to acquire rewards. We selected LCS as the agent architecture for our experiments, because (1) it can contain # in the conditions, and (2) merging knowledge can be simply achieved by merging rule sets. We employed the Zeroth level Classifier System (ZCS)[6] which is a type of LCS.

We did not use the Genetic Algorithm mechanism of ZCS, for the over-generalization problem. That is a fault with ZCS in that too many #s are made in rules and the learning process collapses. We alternatively prepared all rules that could possibly exist. Accordingly, rules which have appropriate #s achieve high values.

3.2 Finite State Machines

We used *Finite State Machines* (FSMs) as the environmental models for the experiments. Figure 4 shows an example of an FSM and the relationship between an FSM and an agent.
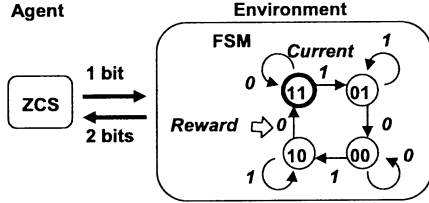
Figure 4: Environment (FSM) and agent (ZCS)

Each FSM in our experiments had four nodes, and each node had two arcs. Each node was assigned two bits, and each arc was assigned one bit. These bits were initialized automatically and randomly. An FSM has a current node. In Figure 4, the current node is "11".

One agent interacts with one environment, i.e. one FSM. Two bits corresponding to the current node are transmitted to the agent. The agent responds to those bits, and one bit corresponding to the agent's action is transmitted to the FSM. The current node is moved in the direction of the arc corresponding to the one bit. This cycle was repeated in the experiments. The reward was transmitted to the agent when the reward arc was used. This reward was used for the agent's

reinforcement learning. In Figure 4, the reward arc is the "0" arc of the "10" node.

4 Experiment

4.1 Experimental design

We conducted experiments to evaluate our index of degrees of confusion. The experimental framework is shown in Figure 5. First, FSMs A and B were automatically prepared, and Agents A and B implemented by ZCS were initialized, then, Agents A and B were applied to FSMs A and B, respectively. In these learnings, the agents acted 10,000 times. We found this number to be adequate for the convergence of learning in our preparatory experiments. After this, the knowledge (i.e. rule sets with rule values) of Agents A and B were just merged and this merged knowledge was embeded into Agent C. Agent C interacted with FSMs A and B.

One hundred sets of FSMs A and B were generated. The numbers of rewards Agent C acquired from FSMs A and B during one hundred actions were recorded, and the performance is the average of two numbers. In addition, this performance was averaged over different one hundred sets of Agents A and B.
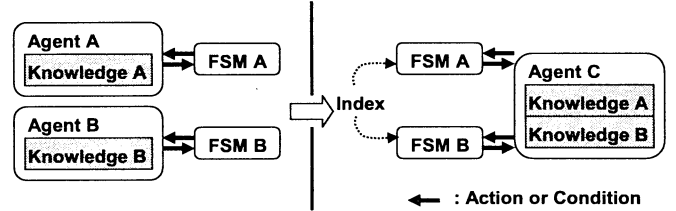
Figure 5: Schema for experiment

4.2 Experimental results

Figure 6 shows the experimental results. The vertical axis shows the performance and the horizontal one shows the index for the degree of confusion. The results for each pair of FSMs are plotted. It can be seen that the smaller the index, the bigger the performance.

5 Discussion

5.1 Effectiveness of index

The index spreads from '0' to '$\frac{28}{3}$'. In the environments of index '0', all extracted rule actions are identical in two extracted rule sets. Thus, agents only need to have a rule whose condition is '##' and action is '0' or '1', to obtain get high performance. If there
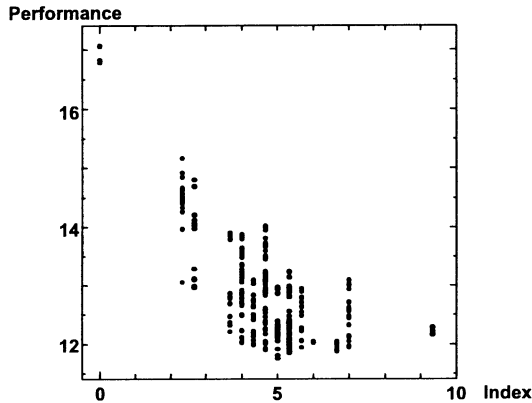
Figure 6: Relationship between performance and index

is a rule which has a different action in two extracted rule sets, the index of those environments is '$\frac{7}{3}$', and the agent has to have a more complex rule set than the one in index '0'. As extracted rules with different actions increase in two extracted rule sets, the index gradually increases, and agents have to have complex rules. Thus, performance decreases monotonously.

The experimental results indicate that the index can estimate the performance of the agent. This means there is the possibility of agents exchanging or substituting knowledge using degrees of confusion.

This index, however, has a problem with accuracy. Figure 6 shows many plots around the mean value of the index range, and their performance is widely spread across diffrent values. For example, the environment of index '$\frac{14}{3}$' may be more confusing than one of index '5'. Thus, there is room for improvement in the index. This improvement will be achieved by adjusting the formula of the index calculation.

5.2 Approach for autonomous measurement

The ultimate goal of our research was for two agents to autonomously measure degrees of confusion. Since the index which we proposed in this paper was based on the confusion of the environments itself, this index has to be improved by taking the values of rules into account.

If agents can autonomously measure degrees of confusion, agents can exchange knowledge effectively. Before agents exchange knowledge, agents can select other agents using the index. For example, one agent calculates the indexes against all the other agents, and selects the agent which has the smallest index. In the experiments in this paper, an agent accepts all rules of two rule sets. If an agent repeatedly exchanges knowl-

edge in this way, there is an explosion in the number of the agent's rules. Thus, some other strategy is necessary. For example, the agent discards half the worthless rules, and accepts the other agent's worthwhile rules. In addition, the method of substitution is similar to that of exchange.

6   Conclusion

This paper proposes an index for degrees of confusion between knowledge that is acquired by two agents implemented by Learning Classifier Systems. This index can estimate the degrees of confusion of knowledge which has generality, and it can be used when agents select the best partner for exchange and substitution in a multi-agent system. We showed that this index has great potential in estimating degrees of confusion through experiments employing agents implemented by Learning Classifier Systems and environments described by FSMs. Future research includes (1) the search for an index which is calculated from agents' knowledge themselves, and (2) experiments in multi-agent systems with three or more agents.

References

[1] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[2] R.S. Sutton, "Reinforcement Learning Architecture for Animats", *The 2nd International Conference on Simulation of Adaptive Behavior (SAB '90)*, pp. 288-296, 1990.

[3] K. Takadama, K. Hajiri, T. Nomura, K. Shimohara, and S. Nakasuka, "Organizational Learning Model for Adaptive Collective Behaviors in Multiple Robots", *Advanced Robotics (The International Journal of the Robotics Society of Japan)*, VSP, Vol. 12, No. 3, pp. 243-269, 1998.

[4] M. Tan, "Multi-agent Reinforcement Learning: Independent vs. Cooperative Agent", *The 10th International Conference on Machine Learning (ICML '93)*, pp. 330-337, 1993.

[5] G. Weiss, "Distributed Artificial Intelligence Meets Machine Learning - Learning in Multi-Agent Environments -", *Lecture Notes in Artificial Intelligence*, Volume 1221, Springer-Verlag, 1997.

[6] S. Wilson, "ZCS: A Zeroth Level Classifier System", *Evolutionary Computation* Vol. 2, No. 1, pp. 1-18, 1994.

DOM/XML-Based Portable Genetic Representation of Morphology, Behavior and Communication Abilities of Evolvable Agents

I. T. Tanev

ATR Human Information Science Laboratories,

2-2-2 Hikaridai, "Keihanna Science City", Kyoto 619-0288, Japan

e-mail: i_tanev@atr.co.jp

Abstract

This document presents the result of our work on the role of genetic representation in facilitating quick design of efficiently running offline learning via genetic programming (GP). An approach of using the widely adopted DOM/XML standard for representation of genetic programs and off-the-shelf DOM-parsers with build-in API for manipulating them is proposed. The approach features significant reduction of time consumption of usually slow software engineering of GP and offers a generic way to facilitate the reduction of computational effort by limitation of search space of genetic programming via handling of only semantically correct genetic programs. The concept is accomplished through strongly typed genetic programming (STGP), in which the use of W3C-recommended standard XML schema is proposed as a generic way to represent and impose the grammar rules in STGP. The ideas laid in the foundation of the proposed approach are verified on the implementation of GP for evolving social behavior of agents in predator prey pursuit problem.

Keywords: genetic programming, XML, phylogenetic learning, agents.

1 Motivation

Developing mobile autonomous robots' controllers is often performed as a sequence of simulated off-line design (phylogenetic learning) of the robot's software model followed by on-line adaptation (ontogenetic learning) on the physical robot situated in real environment [6][8]. Justification to incorporate off-line software simulation into the process of robots controller design comes from the facts that verification of robot behavior on physical robots is extremely time consuming and often dangerous for the robot and surrounding environment. These arguments in favor of software simulation of controller as a process, which precedes the physical design and online learning on real robot become even more relevant within the context of recently emerged trends in robotics such as investigating the social behavior, the role and value of communication in emergent coherence, cooperation and collaboration in the robots' societies situated in inherently competitive or cooperative environments. Addressing these issues directly on the societies, consisting of many of physically built robots seems to be a practically infeasible approach. Simulating robot controllers as software agents and focusing on the model of their relevant features is viewed as a promising way to address the above mentioned shortcomings of direct verification of behavior on physical robots.

GP, which we propose as an approach for offline learning implies that the agent's code is automatically designed by computer system via simulated evolution employing selection and survival of the fittest in a way similar to the evolution of species in the nature. While for many tasks handcrafting the agent code can be seen as natural approach, it might be unfeasible for most of real-world problems due to their typically enormous complexity. Moreover, in many problems the challenge is to develop a solution, which is competitive or even better than human-designed one. Such a solution might be well beyond the abilities of human to handcraft it.

The software model of the evolvable robot's controller should fulfill the basic requirements of being adequate, fast running, and quickly developed [6]. Considering the adequacy of the model as beyond the scope of this document, we intend to highlight the issues related to the efficiency of the system (in terms of reduced developing- and execution time) for evolving agent's behavior. The typically slow developing time of GP stems from the highly specific semantics of main attributes of GP (representation, initial population, genetic operations and fitness evaluation) and the lack of generic support to these attributes in 3G algorithmic languages and corresponding software engineering standards. *Developing time* of GP can be significantly reduced incorporating commodity-off-the-shelf software components and standards in software engineering of GP. The *runtime* of GP can be reduced as a cumulative result of reduced computational effort (the amount of individuals that should be processed in order to obtain a solution with specified probability) and increased computational performance (the amount of individuals evaluated per unit of time). The *objective* of our research is to develop a genetic representation of the evolvable autonomous agents, which based on commodity-off-the-shelf software components and widely adopted industrial standards, would facilitate the achievement of easy and quick development phase of GP and would contribute to the achievement of better computational effort. The long-term aim is to employ such a representation in our research on the emergence and the survival value of social behavior and communication in multi-agent systems.

The remaining of the document is organized as follows. Section 2 introduces GP as algorithmic paradigm for offline learning of agents. It also elaborates the proposed approach of representing evolvable agents (genetic programs) as DOM-parsing trees and discusses its design-time and runtime-related implications. Section 3 presents the result of verification of our approach on predator-prey pursuit problem. Conclusion is drawn in Section 4.

2 Approach

2.1 Algorithmic Paradigm

We consider a set of stimulus-response rules as a natural way to model the reactive behavior of autonomous agents [5] which in general can be evolved using artificial neural networks, genetic algorithms, and GP. GP is a domain-independent problem solving approach in which a population of computer programs (individuals) is evolved to solve problems [7]. The simulated evolution in GP is based on the Darwinian principle of reproduction and survival of the fittest. In GP individuals are represented as parsing trees whose nodes are functions, variables or constants. The nodes that have sub-trees are non-terminals - they represent functions where the sub-trees represent the arguments to function of that node. Variables and constants are terminals – they take no arguments and they always are leaves in the parsing tree. The set of terminals for evolving agent's behavior includes the relevant stimuli such as perceptions (e.g. distance and visible angle to various objects in the world), its own state, etc.; and the response (actions) which the agent is able to perform. The most relevant functions (non-terminals) are the arithmetical and logical operators (>, <, =, +, - , etc.), and the IF-THEN function, establishing the relationship between certain stimulus and corresponding response. A sample stimulus-response rule is shown in Figure 1. It expresses a reactive behavior of turning to the bearing (angle) of the peer agent (Peer_a) plus 10 (degrees) as a result of stimulus of distance to that agent (Peer_d) being less than 20 (mm). A parsing tree of rule, depicted in Figure 1 is shown in Figure 2.

```
IF (Peer_d<20) THEN Turn(Peer_a+10)
```

Figure 1. Sample stimulus-response rule governing the agent behavior.
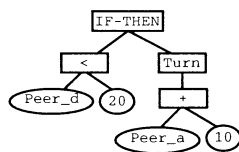
Figure 2. Sample stimulus-response rule represented as a parsing tree in GP.

Parsing-tree representation and its flat equivalents (LISP S-expression, postfix or prefix notations) are typically maintained and manipulated by GP-systems in a customized way. Therefore, an eventual tailoring of the available general purpose GP-systems for the task of evolving autonomous agents would be a time consuming approach. Design of GP-system from scratch would be (if not the only feasible) at least faster and more flexible way, providing that the implementation of main attributes of GP is based on widely adopted industrial standards and off-the-shelf technologies.

2.2 DOM/XML-based Representation of Genetic Programs

Inspired by flexibility and recently emerged widespread adoption of document object model (DOM) and extensible markup language (XML), we propose an approach of representing genetic program as a DOM-parsing tree featuring corresponding flat XML text. Our additional inspiration comes from the fact that despite of the reported use of DOM/XML for representing computer architectures [10], source codes [11], and agents' communication languages [4] we are not aware about any attempts to employ this technology for representing evolvable structures such as genetic programs in generic, standard, and portable way. Our approach implies performing genetic operations on DOM-parsing tree using off-the shelf, platform- and language neutral DOM-parsers, and using XML-text representation as a format, feasible for migration among the computational nodes in eventual distributed GP.

The DOM-parsing tree of the sample rule considered earlier (Figure 1) looks in exactly the same way as parsing tree in canonical representation of genetic programs (Figure 2). However the flat representation of the rule is XML (Figure 3), rather than LISP S-expression.

```
<GP>
  <IF-THEN>
    <LE>
      <PERC>Peer_d</PERC>
      <PERC>20</PERC>
    </LE>
    <TURN>
      <PLUS>
        <PERC>Peer_a</PERC>
        <PERC>10</PERC>
      </PLUS>
    </TURN>
  </IF-THEN>
</GP>
```

Figure 3. XML representation of stimulus-response rule in GP.

2.3 Design-time Implications

In contrast to the typical approaches to manipulate parsing trees using custom code and representations, DOM-based approach offers benefits of requiring *minimum programming efforts* and allowing developers to use the software platform, developing language, and/or programming paradigm which better fits the aims of concrete implementation of GP. These benefits are result of:

- Use of API of DOM-parsers: parsing tree of genetic program is manipulated using built-in API of DOM-parsers. There is no need to develop, test and debug such API,
- Platform neutrality of parsers: DOM-parsers are available for virtually any of widely used software platforms (e.g., as Java classes), facilitating the portability of GP across different software platforms,
- Language neutrality of parsers: DOM-parsers are also available as language-neutral components (e.g. Microsoft COM), offering the same programming model of parsing trees regardless of the language employed to develop the code of GP that manipulates them, and
- Paradigm neutrality of parsers: DOM-parsers are available for programming paradigms, such as data-

base stored procedures, implemented on programming languages designed mainly to manipulate database data and lacking the adequate syntax and semantic support of main attributes of GP. Proposed approach is viewed as the only way to implement an efficient GP directly on the database server, minimizing the client-server data-transmission overhead in real-world multi-tiered, web-based applications.

2.4 Run-time Implications

The potential strength of GP to automatically evolve a set of stimulus-response rules featuring arbitrary complexity without the need to a priory 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 optimal solution to the problem. A well-known way to limit the search space, and consequently, to reduce the computational effort of GP is to impose a restriction on the syntax of evolved genetic programs based on their a priory known, domain specific semantics. The approach is known as strongly typed genetic programming (STGP) and its advantage over canonical GP in achieving better computational effort is well proven [9]. Our objective is to propose a generic way to support STGP, consistent with proposed DOM/XML representation of genetic programs.

Considering the same sample rule as shown in Figure 3, and multimodality of perception information, it is noticeable that all nodes (i.e. both the functions and their operands) are associated with *data types* such as distance (Peer_d, 20), angle (Peer_a, 10), Boolean (Peer_d<20), etc. An eventual arbitrary creation or modification of such genetic program semantically would make little sense: e.g. it is unfeasible to maintain a rule with stimulus-related part featuring arithmetical expression involving operands of different data types (e.g. distance, angle and Booleans) at least because physically they are of different dimensions. In addition, there is clear possibility in maintaining introns if such an expression compares perception variable with constant beyond the range of corresponding sensor (e.g. Peer_d>1000, in case that sensor range is 400). Analogically, the semantics of action Turn() implies a parameter of corresponding data type - an angle. However data types are not explicitly specified in considered so far DOM/XML representation of genetic program, and consequently, not evident for the routines that create and alter it. To address the issue, we explicitly introduced the notion of type (in a way similar to STGP) represented as XML-tag for all the entities the genetic programs are composed of. In addition, we established a set of rules (i.e. grammar) describing the allowed relationship between types in semantically meaningful genetic program. Routines that create and alter genetic programs (e.g. creation of initial population and mutation) refer to the type of entity they are going to currently alter and impose the corresponding constraints to the sub-tree structures or values being generated. In crossover only nodes (with corresponding sub-trees) of the same type can be swapped.

In our approach we represent the grammar of STGP

as a *XML schema* [1][3], which is an official World Wide Web Consortium (W3C) recommended, standard way to define the relationship among the entities in XML-document (i.e. genetic program). Since the syntax of schema conforms to the XML-standard, the routines those create and alter the parsing tree of genetic program access the schema via API of DOM-parsers in a way, identical to the way of accessing DOM/XML-based representation of genetic program.

We developed a set of formal rules for translating a BNF-defined grammar of STGP into corresponding XML-schema. Without touching the details of such rules, we present the resulting fragment of XML-schema (Figure 4) that corresponds to stimulus-related part of sample rule illustrated in Figure 3. Notice the definition of the sensory abilities – the *morphology* of the agent: the kind of perception information (Wall_d, Peer_d) and the range of corresponding sensor (0..400). In a similar way XML schema defines the response abilities (actions) of the agents. Considering the communication as response for the speaker and stimulus for listener, the XML-schema offers a generic way to define the *communication abilities* of the agent. The section of XML-representation of strongly typed genetic program created applying the same fragment of XML schema is shown in Figure 5.

```
<xs:complexType name="IF-THEN"><xs:sequence>
  <xs:element name="COND-THEN" type="COND-THEN" />
  <xs:element name="THEN" type="THEN" />
</xs:sequence></xs:complexType>
<xs:complexType name="COND-THEN">
  <xs:choice>
  <xs:element name="COND_TDist" type="COND_TDist" />
  ...
</xs:choice></xs:complexType>
<xs:complexType name="COND_TDist">
  <xs:sequence>
  <xs:element name="VAR_TDist" type="VAR_TDist" />
  <xs:element name="OPER_TDist" type="OPER_TDist" />
  <xs:element name="CONST_TDist" type="CONST_TDist" />
</xs:sequence></xs:complexType>
<xs:simpleType name="VAR_TDist">
  <xs:restriction base="xs:string">
  <xs:enumeration value="Wall_d" />
  <xs:enumeration value="Peer_d" />
</xs:restriction></xs:simpleType>
<xs:simpleType name="OPER_TDist">
  <xs:restriction base="xs:string">
  <xs:enumeration value="GE" />
  <xs:enumeration value="LE" />
</xs:restriction></xs:simpleType>
<xs:simpleType name="CONST_TDist">
  <xs:restriction base="xs:integer">
  <xs:minInclusive value="0" />
  <xs:maxInclusive value="400" />
</xs:restriction></xs:simpleType>
```

Figure 4. XML schema defining the grammar corresponding to the stimulus-related part of rule shown in Figure 1 and Figure 2.

3 Verification

In order to verify the feasibility of our concept of representing genetic programs as DOM parsing tree and its design- and run-time implication we implemented a prototype of GP system for offline phylogenetic learning of agents in predator-prey pursuit problem [2]. The problem comprises four predators (agents) whose goals are to capture a prey by surrounding it on all sides in a world.

In our work we assume that predator can see the prey and the closest predator-peer only if they are within the range of visibility of its simulated (covering an area of 360 degrees) sensors. Prey employs optimal escaping strategy as soon as predator becomes "visible". The maximum speed of prey is higher than the maximum speed of predator. The world is two-dimensional continuous torus, where the only way to capture the prey is to surround it.

```
<GP>
  <IF-THEN>
    <COND-THEN>
      <COND_TDist>
        <VAR_TDist>Peer_d</VAR_TDist>
        <OPER_TDist>LE</OPER_TDist>
        <CONST_TDist>20</CONST_TDist>
      </COND_TDist>
    </COND-THEN>
    <THEN>
    ...
    </THEN>
  </IF-THEN>
</GP>
```

Figure 5. XML-representation of strongly typed genetic program, created applying the fragment of XML schema shown in Figure 4.

GP system runs on W2K system and employs Microsoft DOM parser – MSXML4.0. Considering the elaboration of the issues related to the emergence of surrounding behavior as irrelevant to the aims of this document and focusing on the verification of proposed approach as technology for representing genetic programs, we would like to summarize the result as follows:
- Developing the prototype of GP is significantly alleviated by use of DOM-parser. We measured about few [men x days] of development efforts (without considering the development of user interface), and
- XML schema offers generic way to impose the semantics constraints of genetic programs represented as DOM-parsing trees. The resulting computational effort for considered instance of predator-prey problem is relatively low and is in order of several thousands evaluations of genetic programs.

4 Conclusion

We presented the result of our work on the role of genetic representation in facilitating quick design of efficiently running offline phylogenetic learning via GP. We proposed a portable representation of evolvable agents (genetic programs) based on widely adopted DOM/XML standard. The manipulation of genetic programs implies the use of build-in API of off-the-shelf DOM-parsers. The approach features significant reduction of time consumption of usually slow process of software engineering of GP. In addition it offers a generic way to facilitate the reduction of computational effort via limitation of search space of GP by handling of only semantically correct genetic programs. Consistent with the concept of strongly typed GP, an approach of using W3C-recommended standard XML schema is developed as a generic way to represent and impose the grammar rules. The ideas laid in the foundation of the proposed approach are verified on the implementation of genetic programming for evolving social behavior of agents in

evolving social behavior of agents in predator prey pursuit problem. Due to the domain neutrality of GP, the approach can be applied for quick developing of efficiently running GP in various problem domains.

Acknowledgements

References

[1] D.Beech, M.Maloney, N.Mendelsohn, H. Thompson, XML Schema Part 1: Structures. W3C Recommendation, 2001. http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

[2] M. Benda, V. Jagannathan, and R. Dodhiawala, On optimal cooperation of knowledge sources - an empirical investigation. *Technical Report BCS-G2010-28*, Boeing Advanced Technology Center, Boeing Computing Services, Seattle, Washington, 1986.

[3] P.Biron, A.Malhotra, XML Schema Part 2: Datatypes W3C Recommendation, 2001, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

[4] B.N.Grosov and Y.Labrou, An Approach to using XML and a Rule-based Content Language with an Agent Communication Language , *IBM Research Report*, RC 21491 (96965), 1999.

[5] J.H.Holand, *Hidden Order: How Adaptation Builds Complexity*, Cambridge, MA: Helix Books, 1996.

[6] N.Jacobi, Minimal Simulations for Evolutionary Robotics. Ph.D. thesis, School of Cognitive and Computing Sciences, Sussex University, 1998.

[7] J.R.Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Cambridge, MA: MIT Press, 1992.

[8] L.Meeden and D.Kumar, Trends in Evolutionary Robotics, *Soft Computing for Intelligent Robotic Systems*, edited by L.C. Jain and T. Fukuda, Physica-Verlag, New York, NY, pp.215-233, 1998.

[9]. D.J.Montana, Strongly Typed Genetic Programming, *Evolutionary Computation*, Vol.3, No.2, pp.199- 230, 1995.

[10] The Open Group, Architecture Description Markup Language (ADML): The new XML-based standard for IT architecture interoperability, 2001. http://www.opengroup.org/tech/architecture/adml/adml_home.htm

[11] Y.Zou, K.Kontogiannis, Towards A Portable XML-based Source Code Representation, *Proceedings of ICSE 2001 Workshops of XML Technologies and Software Engineering (XSE 2001)*, Toronto, Canada, May 15, 2001.

Optimized Space Search by Distributed Robotic Teams

Peter Sapaty   Masanori Sugisaka

Department of Electrical and Electronic Engineering
Oita University, Dannoharu 700, Oita 870-1192, Japan
+81-97-554-7831, +81-97-554-7841 (fax)
{p-sapaty, msugi}@cc.oita-u.ac.jp, psapaty@hotmail.com

Abstract

The paper describes optimized solutions for the distributed space search and processing by organized robotic teams working in WAVE-WP technology. The latter provides for efficient integration of physical and virtual worlds (PW and VW) within the same model, where optimized solutions can first be developed in VW, and then subsequently or simultaneously transformed into activities in PW. Many examples of the related WAVE-WP code are given, proving its expressive power and compactness.

Keywords: Distributed search, robotic teams, WAVE technology, spatial scenarios, parallel interpretation.

1 Introduction

Many problems potentially requiring distributed robotics are linked with parallel search of large physical spaces. De-mining territories, looking for minerals (domestic or on other planets), patrolling regions, cleaning territories, robotic fishery, and disaster relief may be a few of them. Problems like de-mining, for example, may need an absolute guarantee that the whole territory is thoroughly examined.

One of the main problems here is that it is difficult, often impossible, to keep a central database of the space to be searched, as most of local information is highly dynamic and changeable at runtime, and can be kept and constantly updated in local brains of robots only. Any non-local system solutions must take this into account, and must themselves be effectively distributed.

Also, in dynamic and hostile environments, it is not possible to make any optimization and scheduling in advance, as time of search and processing of parts of space may depend on many, emergent, circumstances, and individual robots may also fail or can be destroyed indiscriminately at run time.

Traditional approaches to distributed system control and optimization are analytical: They first define parts, as "agents", their interconnections and inter-control, and then try to comprehend and manage the whole. This often does not work properly, especially when there are many parts, and their interactions are emergent or unsafe. Also, existing approaches usually offer only partial solutions for the multi-robot control problems [1].

In the current paper, we propose a general solution for this problem, based on a universal parallel and fully distributed model and technology WAVE-WP (or World Processing), previously described in [2-4]. Using this technology, the team of robots can be effectively converted into a universal and parallel spatial computer [4] capable of solving any optimization problems autonomously, and without any central resources.

2 The WAVE-WP technology

WAVE-WP provides an opportunity to program the overall system behavior not as a collection of interacting agents, as usual, with known difficulties of achieving the needed global states, but as an organized whole, concentrating rather on the overall system integrity, efficiency, and pursuit of global goals.

The technology implants a universal knowledge processing, command and control module, interacting with other such modules, into main components of the distributed system to be controlled. A stationary or mobile network of such modules (or "wave interpreters", WI) cooperatively executes a high-level scenario language, dynamically creating and spatially processing both virtual and physical worlds, as shown in Fig. 1.
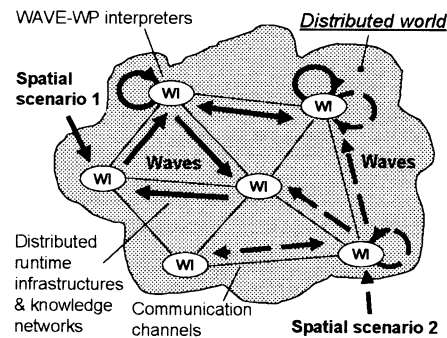
Fig. 1. Main elements of the WAVE-WP technology

Parallel spatial scenarios (or "waves") in WAVE-WP, determining the overall system behavior, can be injected from any WI, and waves can compete or cooperate while navigating distribute systems. Both hierarchical and horizontal recursive control can emerge, fully mobile too. Distributed knowledge networks that can freely migrate

between interpreters are created and shared by different navigation processes. Task-oriented runtime infrastructures are formed dynamically, supporting the distributed language interpretation and movement of program code and data. Multi-scenario operation and multi-user access are key features too. WI may be installed in hosts of a computer network or in mobile robots, causing the latter move and process and exchange both information and physical matter or objects, while making collective decisions autonomously.

Waves have the following syntax, see also [3,4]:

wave → { *advance* ; }
advance → { *move* , }
move → *constant* | *variable* | { *move act* } |
(*wave*) | [*wave*] | *rule* (*wave*)
variable → *nodal* | *frontal* | *environmental*

Words in italics represent syntactic categories, braces show zero or more repetitions of a construct with a delimiter at the right, and vertical bar separates alternatives. The semicolon allows for sequential, while comma for parallel invocation of program parts, and round and square brackets are used for structuring.

Successive program parts, or *advances*, develop from all nodes of the set of nodes reached (SNR) by a previous advance, whereas parallel or independent parts, or *moves*, constituting advances, develop from the same nodes, splitting processes and adding their SNRs to the resultant SNR of the advance.

Moves can point at a resulting value directly (as a *constant* or *variable*). They can also form complex space navigating and data processing expressions consisting of arbitrary moves separated by elementary operations (*acts*), with the return of (local or remote) results for further processing. Acts represent data processing, hops in both physical and virtual spaces, and local control.

Moves can themselves be arbitrary waves, optionally prefixed by control *rules*. The latter establish non-local constraints and contexts over space-evolving waves, like the ability of creating networks, also allowing WAVE-WP to be used as a conventional language.

Variables, called spatial, can be of three types: *nodal* associated with virtual of physical nodes and shared by different waves, *frontal* propagating with waves, and *environmental* accessing elements of internal and external environment navigated by waves.

This recursive language structure provides for very compact expression of complex space navigation, data processing and control operations which can be carried out in fully distributed and parallel mode.

3 Creating the distributed virtual world

We will be using traditional representation of the search space by a set of polygons [1], as shown in Fig.2, where each polygon is assigned parameters describing its dimensions and shape, sufficient for navigation.
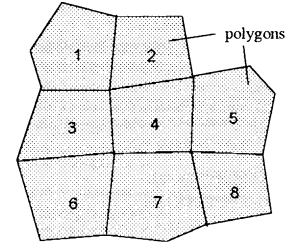
Fig. 2. Polygons representing space to be searched

This set of polygons together with their connectivity graph (CG) can be represented in WAVE-WP in an integral form, as a knowledge network (KN), shown in Fig. 2. The polygon data will exist as p i nodes (keeping strings of polygon parameters), attached to corresponding CG nodes i by links p, with g naming CG links.
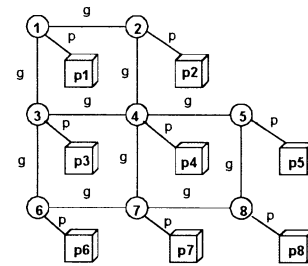
Fig.3. Representation as a knowledge network

A variety of possibilities exist in WAVE-WP for an efficient expression and creation of such a network in a distributed environment. Most compact ones use a parallel depth-first spanning tree template [3] shown in Fig. 4 by bold oriented arcs, with steps numbered at them (same numbers identifying potentially parallel steps).
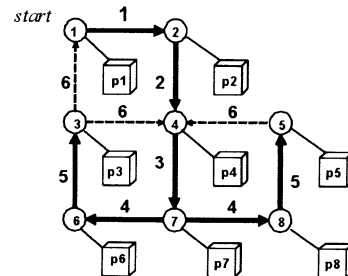
Fig.4. A depth-first spanning tree template

The WAVE-WP program will be as follows (with # being a hop act, and ## hopping to the already existing nodes):

```
create(
(@#1;[p#p1];g#2;[p#p2];g#4;[p#p4];g#7;[p#p7];
(g#6;[p#p6];g#3;[p#p3];g##(1,4)),
(g#8;[p#p8];g#5;[p#p5];g##4))
```

It can start from any robot, and the KN will be automatically distributed between their brains in an optimum way, where some links may happen to connect

nodes in different robots. If needed, distribution between particular robots can be set up explicitly:

```
create(
R=robot.1;@#1;[p#p1];g#2;[p#p2];
R=robot.3;g#4;[p#p4];g#7;[p#p7];
(g#6;[p#p6];R=robot.1;g#3;[p#p3];g##(1,4)),
(R=robot.2;g#8;[p#p8];g#5;[p#p5];g##4))
```

which uses environmental variable RESOURCES (or R for short). The resultant distribution of KN between robots is shown in Fig. 5.
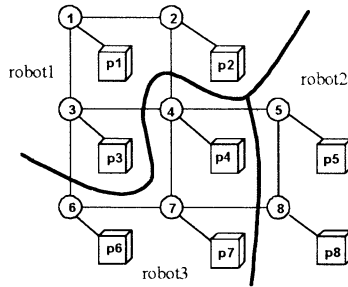
Fig.5. Network distribution between robots

4 Parallel space search and processing

A variety of space search and processing tasks can be effectively solved in this virtual world, represented as KN, in parallel and fully distributed mode. The solutions found on the KN cooperatively can be subsequently or simultaneously converted into physical movement of robots, with these phases may overlap in time and space as one integral process. We will consider hrinafter only some elementary program examples, with orientation on space cleaning applications [1], and on different levels of abstraction.

4.1 Most general descriptions

On top level we may write in WAVE-WP just what we generally want: To reach and clean all polygons simultaneously and, say, only once. The program will be as follows:

```
@#(1,2,3,4,5,6,7,8);p#?clean
```

The invoked external procedure clean will use polygon parameters accessed from corresponding CG nodes by links p. Another desire may be a simultaneous and repeated cleaning of every polygon with some interval, say an hour, while staying in the polygons indefinitely, after reaching them only once. Using environmental variable TIME (or T, for short), this will look like:

```
@#(1,2,3,4,5,6,7,8);repeat(p#?clean;T+=360)
```

If we want a repeated simultaneous cleaning of all polygons with revisiting them each time (say, using robotic resources elsewhere during the breaks), the program will be as follows:

```
repeat(qt(@#(1,2,3,4,5,6,7,8);p#?clean),T+=360)
```

The distributed brain of the robotic group will be dynamically appointing available robots to polygons, so actual parallelism will depend on the dynamic availability of mobile hardware resources. The quality of the shared use of robots will entirely depend on the internal WAVE-WP distributed interpretation mechanisms, which may not always be optimal.

4.2 Local multiple branch search

We may assist the distributed WAVE-WP interpreter in making hardware distribution decisions by providing a more detailed solution of the problem, say, as a limited set of competing branches. Each branch will move between, and clean, the non-cleaned polygons, blocking other branches from visiting the occupied polygons.

With single visiting of polygons, and moving to neighboring polygons only, while checking the state of neighbors sequentially, the program, consisting of three branches starting from polygons 1, 4, and 7, will be:

```
@#(1,4,7);M=1;p#?clean;
repeat(or(g#;M==nil;M=1);p#?clean)
```

where nodal variable M marks polygons as cleaned in the corresponding CG nodes. Each branch terminates if there are no neighboring non-cleaned polygons. A quicker solution will be possible if all neighboring polygons are checked in parallel, with the first replying "non-cleaned" to be the next step:

```
@#(1,4,7);M=1;p#?clean;
repeat(orparallel(g#;M==nil);M=1;p#?clean)
```

These solutions, based on searching neighboring polygons only, may, however, fail to clean all polygons. Another variant may be a regular random movement to empty neighboring polygons, with repeated visiting and cleaning polygons, and endless attempts to find an empty neighbor:

```
@#(1,4,7);M=1;p#?clean;
repeat(or(random(g#;M==nil;M=1;(BACK#;M)=nil;
       p#?clean),))
```

If cleaning of polygons is still required only once, with revisiting of polygons allowed, the previous program can be easily modified for this, as follows (using additional nodal variable N):

```
@#(1,4,7);M=2;p#?clean;
repeat(or(random(g#;M!=2;N=M;M=2;(BACK#;M)=1;
       or((N==nil;p#?clean),)),))
```

The above two solutions guarantee that all polygons will sooner or later be cleaned. We may also introduce a more expensive global search for a non-cleaned empty polygon, not necessarily neighboring one, which will always guarantee a complete solution. Robots may, however, happen to move to the destination polygons through the occupied ones, so collision avoidance is to be present at the implementation level. The program code will be:

```
@#(1,4,7);M=1;p#?clean;
repeat(or(@#;g#!=nil;M==nil;M=1;p#?clean))
```

In a global hop to all other CG nodes, the latter are distinguished from the polygon nodes as having incident g-type links.

A combination of local and global search, where global search is applied only if local fails, may be useful too. The following example uses sequential search of neighbors from one of the previous examples, with endless search for a suitable next polygon:

```
@#(1,4,7);M=1;p#?clean;
repeat(or(or(g#;M==nil),or(@#;g#!=nil;M==nil));
      M=1;p#?clean)
```

The assignment of robots to branches will be done by the distributed interpreter automatically. But we may also explicitly assign particular robots to the branches, as follows:

```
(@#1;Fr=robot.3),(@#4;Fr=robot.2),
(@#7;Fr=robot.1);M=1;(p#,Fr)?clean;
repeat(or(or(g#;M==nil),or(@#;g#!=nil;M==nil));
      M=1;(p#,Fr)?clean)
```

Keeping the robot's name in personal frontal variable Fr, each branch will always drive through the physical space only the robot dedicated to it, despite possibly propagating electronically through other robot's brains holding the currently navigated virtual world partition (so robots may actually happen to give command to each other to move and clean).

The initial distribution of robots between polygons and the cleaning activity in them by this program is shown in Fig.6. Occupation of polygons is reflected by shading their CG nodes, with cleaned polygons having their nodes (boxes) shaded too.
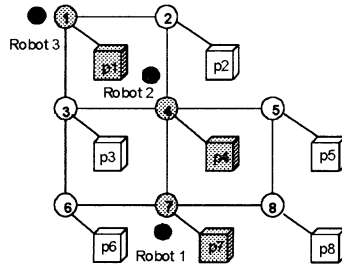
Fig.6. Initial distribution of robots

4.3 Full depth search for polygons

We have considered a single-depth search for empty polygons. Much better solutions can be achieved by a full depth search for each robot, with finding optimum paths to the empty and non-cleaned polygons.

The following program first creates a breadth-first spanning tree from a position occupied by the robot, avoiding occupied polygons and terminating in the first met non-cleaned and empty polygons. Then the closest such polygon is chosen as the next one to clean, and the found path to it is used to drive the robot to the destination, possibly through other, empty, polygons.

```
(@#1;Fr=robot.3),(@#4;Fr=robot.2),
(@#7;Fr=robot.1);F=Fd;M=2;(p#,Fd)?clean;N=1;
```

```
repeat(
  F=min(
    repeat(g#;M!=2;N==nil;N=1;Fp&=C;Fl+=1;
          or((M==nil;Fl&Fp!done),));nil);
    F!=nil;N=nil;M=1;
    repeat(F:1=nil;F!=nil;g#F:1;(p#,Fd)?move);
    M=2;(p#,Fd)?clean)
```

This program, for example, will find a path 2-4-7-8 to non-cleaned polygon p8 for robot 3 staying in p2, as shown in Fig. 7, subsequently driving the robot to p8 via the waypoints found.
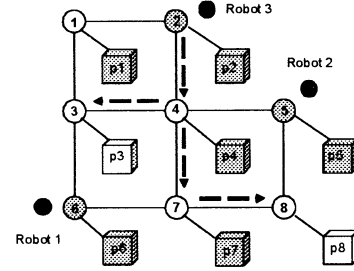
Fig.7. A spanning tree search for unvisited polygons

This may not be a full solution too. But if we allow to develop the spanning tree via the occupied polygons, not only free ones, the solution of finding paths to non-cleaned polygons will be universal, with the potential danger of collisions, however. The above program can do this with elementary modifications, treating, for example, the path 2-4-5-8 for robot 3 (see fig. 7) as legal too. We can also easily express in WAVE-WP a detailed and universal shortest path tree solution where moving through occupied polygons lengthens the path (with a proper length-increasing function chosen), so the paths via empty polygons could be more preferable, if exist in principle. This in WAVE-WP will be as follows:

```
(@#1;Fr=robot.3),(@#4;Fr=robot.2),
(@#7;Fr=robot.1);P=Fr;Fo=2;M=2;(p#,Fr)?clean;
repeat(Nd=0;
  sequence(
    quit(
      [repeat(
        g#;or((M<=1;Fd+=1),Fd+Fo);Nd==nil;Fd<Nd;
        Nd=Fd;Np=(BACK#);or((M=nil;!done),))];
      F=min(repeat(g#;Np==(BACK#);Fp&=CONTENT;
        or((M==nil;Nd&Fp!done),));nil)),
      (F!=nil;M=1;
      repeat(F:1=nil;F!=nil;g#F:1;(p#,Fd)?move);
      M=2;(p#,Fd)?clean)))
```

More on parallel and distributed shortest path solutions in WAVE can be found in [3].

5 Runtime space modification

The proposed organization of multi-robot space search may be dynamic and open, with new search regions and new robots added at runtime, while others removed, in a seamless way. KN can be redistributed between robots at runtime, say, when they are overloaded, fail, or are destroyed. For example, adding new polygons p9 and p10 and linking them, correspondingly, with existing ones p7 and p8, via new

CG nodes 9 and 10, can be done by:

```
create(@#9;[p#p9,g##7];g#10;p#p10,g##8)
```

If we want this new network part to be placed in a particular robot, we should write:

```
R=robot.4;
create(@#9;[p#p9,g##7];g#10;p#p10,g##8)
```

KN nodes can move freely between robots. For exampled,, we can easily move nodes 1 and 3 from robot 1 to robot 3, node 4 from robot 3 to robot 1, and node 7 from robot 3 to robot 2. Links from these nodes to other nodes as well as between them will be saved during the movement. These changes can be done by the program:

```
(@#(1,3);DOER=robot.3),(@#4;DOER=robot.1),
(@#7;DOER=robot.2)
```

We can also remove existing polygons at runtime, like say p6, together with the corresponding CG node 6:

```
@#6;sequence(p#,;CONTENT=nil)
```

All these actions can be done separately or simultaneously, at any time, and starting from the same or from different robots. The resultant distribution of KN between robots for the changes above, taking into account the new nodes added, is shown in Fig. 8.

Changes of the distributed knowledge network may take place simultaneously with any wave processes on it. For example, all the above solutions for robots finding and moving between polygons will work during any runtime modification of the virtual space.
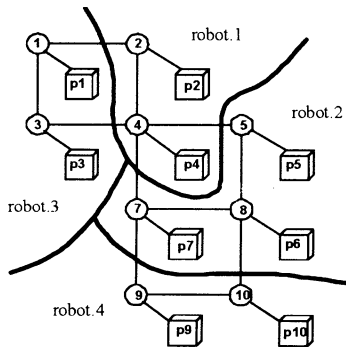
Fig. 8. The resultant distribution of the virtual world

6 Conclusions

We have proposed a fully distributed and universal solution for the distributed space search and processing by robotic teams, based on the WAVE-WP technology. Its key features being as follows.

• The space information, as a virtual world, can be arbitrarily distributed between robots, and robots may physically serve territories not necessarily the same as they keep in their memory. The robots can effectively control each other during the cooperative space search processing.

• The optimized solutions are first found by a group

of robots in the virtual world, distributed between the robot's brains, and then are mapped on the distributed physical world. All group solutions are achieved without any central control.

• The proposed organization is extremely flexible. Depending on the policy of distribution and redistribution of the connectivity graph and polygon data between robots, from fully distributed to fully centralized group control solutions can emerge, within the same application programming.

• The system management scenarios in WAVE-WP are very compact, often tens to hundreds times shorter than in other languages, which makes WAVE-WP language suitable for direct interpretation by robotic teams in both software and hardware.

Many of the presented ideas and solutions had been tested via the Internet [3] in distributed environments, and can be easily embedded into real robots.

Acknowledgments

References

[1] M. Jager, "Cooperating Cleaning Robots", in book Distributed Autonomous Robotic Systems, Springer, ISBN 4-431-70339-X, pp. 237-246.

[2] P.S. Sapaty, "A New Technology for Integration, Simulation, and Testing of Distributed Dynamic Systems", in NATO Proceedings Integration of Simulation with System Testing, RTO-MP-083, AC/323(SCI-083)TP/43, June 2002.

[3] P.S. Sapaty, Mobile Processing in Distributed and Open environments, John Wiley & Sons, ISBN: 0471195723, New York, 1999, 416p.

[4] P. Sapaty, and M. Sugisaka, "Universal Distributed Brain for Mobile Multi-robot Systems", in book Distributed Autonomous Robotic Systems, Springer, ISBN 4-431-70339-X, pp. 434-443.

Learning control of autonomous airship for three-dimensional pursuit problem

Ayako Nishimura[†]   Hidenori Kawamura[†]   Masahito Yamamoto[†]   Azuma Ohuchi[†]

[†]Laboratory of Harmonious Systems Eng., Division of Complex Systems Eng.,
Institute of System and Information Eng., Graduate School of Eng., Hokkaido University
Nishi 8, Kita 13, Kita-ku, Sapporo, Hokkaido, 060-8628, JAPAN
{ayako,kawamura,masahito,ohuchi}@complex.eng.hokudai.ac.jp
http://ses3.complex.eng.hokudai.ac.jp

Abstract

The final goal of our effort is to realize the autonomous robot that moves in three-dimensional space. For this purpose we use small airship as a platform of the autonomous robot. And we choose the pursuit problem for task to develop the autonomous airship because it may apply other tasks. In order to design automatically controlled airship, we make a simulator suits our airship. The airship tends to be influenced by external condition such as air pressure and air temperature, and it has great inertia. We apply *simple Q-learning* designing the control system to realize adaptive control system, and clarify the difficulty of designing the control system using *simple-Q-learning*.

keywords: autonomous robot, simple-Q-learning

1 Introduction

There are a lot of autonomous robot researches in recently, but most of them deal with two-dimensional locomotion. The final goal of this investigation is to realize the autonomous robot moves in three-dimensional space. For this purpose, we use small airship as platform because it is easy to operate among other aircrafts.

We deal with the pursuit problem by two airships. Pursuit problem needs to achieve recognition an opponent airship, spatial awareness, prediction trajectory of an opponent airship, planning own optimum trajectory, and tracking trajectory. The items listed above are also needed in other tasks, for example, flight toward objective point, flight avoiding obstacles, and formation flight. If the airship can pursuit automatically, its technology can be applied other tasks.

The motion of small airship is difficult to be analyzed because it has great inertia and air resistance. Fur-

thermore it has nonholonomic constraint. It is difficult to control even for human. And it tends to be influenced by external condition such as air pressure and air temperature. We apply evolutionary computation and reinforcement learning to design adaptive control system. In particular, we use simulation environment as the pilot study for realization of real robot system in this paper. We apply *simple-Q-learning* to the pursuer.

2 Basic model of Airship

2.1 Structure of our airship
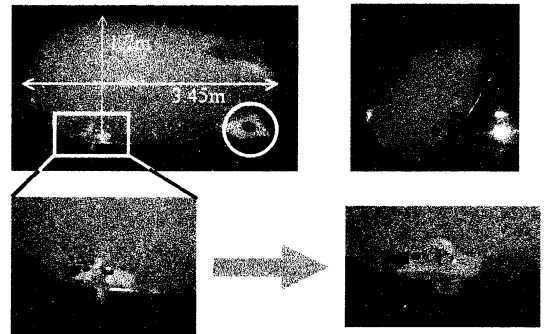
Figure 1: Structure of our airship

The airship has two propulsion units. The rectangular part(gondola propeller) in Figure 1 enables airship to move forward, backward, upward, and downward. The circular part(tail propeller) enables airship to turn. And four tails are attached. The propeller performs right-handed rotation and left-handed rotation. Its propulsion is different. Left-handed rotation is more powerful than right-handed rotation. The length of the airship is 3.45[m] and the maximum diameter is 1.7[m].

2.2 Dynamics

We form dynamic equation referring Zwaan[1] and Gomes[2]. The coordinate system of dynamic equation is {Airship} in Figure 2 which is fixed to the airship. {U} is the coordinate system fixed to the earth. The origin of {Airship} is center of volume. (u,v,w) is velocity vector and (p,q,r) is angular velocity vector, each component of the vector is about x, y, and z axes respectively, and α, β, γ in {U} is angle about x, y, and z axes respectively. The equations of motion for our airship are as follows.
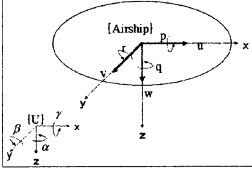
Figure 2: coordinate system
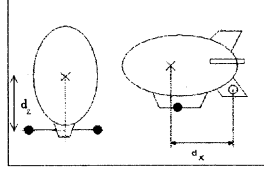
Figure 3: definition of d_x, d_z

X-Z plane

$$x = [\delta u \quad \delta w \quad \delta q \quad \delta\beta]^T$$
$$\tau = [\delta T_{cmn} \quad \delta T_v]^T$$

$$M\dot{x} = Ax + B\tau + F$$

$$M = \begin{bmatrix} m + A_{11} & 0 & ma_z & 0 \\ 0 & m + A_{33} & -ma_x & 0 \\ ma_z & -ma_x & I_y + A_{55} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} -X_u & 0 & 0 & -(mg - f_b) \\ 0 & -Z_w & 0 & 0 \\ 0 & 0 & -M_q & -a_z mg \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 0 & d_z & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}^T$$

$$F = \begin{bmatrix} (m + A_{22})rv + ma_x r^2 \\ 0 \\ (I_{xz} + K_p)r^2 - a_z rv \\ 0 \end{bmatrix}$$

X-Y plane

$$x = [\delta v \quad \delta p \quad \delta r \quad \delta\gamma]^T$$
$$\tau = \delta T_{diff}$$

$$M\dot{x} = Ax + B\tau + F$$

$$M = \begin{bmatrix} m + A_{22} & -ma_z & ma_x & 0 \\ -ma_z & I_x + A_{44} & 0 & 0 \\ ma_x & 0 & I_z + A_{66} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} -Y_v & 0 & 0 & mg - f_b \\ 0 & -K_p & 0 & -a_z mg \\ 0 & 0 & -N_r & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$B = [0 \quad 0 \quad d_x \quad 0]^T$$

$$F = \begin{bmatrix} -(m + A_{11})ur \\ ma_z ur \\ -ma_x ur \\ 0 \end{bmatrix}$$

M is a mass matrix, including added mass and inertia moment. A is a matrix of aerodynamics forces, gravity and buoyancy, B is a propulsion matrix, F is a vector contains centrifugal force and the Coriolis force. For a meaning of each parameter see below.

- m : the airship mass
- g : acceleration of gravity
- f_b : the buoyancy force
- $A_{11} \sim A_{66}$: the added mass direction of x, y, and z axes and the added inertia moment around x, y, and z axes in order
- $X_u, Y_v, Z_w, K_p, M_q, N_r$: the aerodynamics coefficient (nondemensional coefficient) direction of x, y, and z axes and around x, y, and z axes in order
- d_x, d_z : the distance from center of volume to propeller (Figure 3)
- T_{cmn} : the propulsion direction of x axes
- T_v : the propulsion direction of z-axes
- T_{diff} : the propulsion around z-axes

A lot of parameters are included in equations, we found the values of parameters by numerical computation and measuring. The parameters are shown following Table 1. The coordinate system of the above-described equations of motion is {Airship}. So, it needs to transform the {Airship} to {U} via a direction cosine matrix for the simulator.

m	6.700000	f_b	6.700000	d_z	0.950000
d_x	1.360000	a_x	-0.034000	a_z	0.287000
A_{11}	1.320000	A_{22}	4.537000	A_{33}	4.537000
A_{44}	0.000000	A_{55}	1.252000	A_{66}	1.252000
X_u	0.118853	Y_v	2.262096	Z_w	2.262096
K_p	0.237706	M_q	1.100000	N_r	1.100000
I_x	2.833000	I_y	4.750000	I_z	3.684000

Table 1: parameters list

2.3 Simulator

Numerical computations are carried out with the Runge-Kutta method of 0.001[sec] step. The control sampling step is 1.0[sec]. And we use OpenGL library to visualize data.

3 Application of basic model

3.1 Application of reinforcement learning

The pursuer decides its action learning by *simple-Q-learning*. The algorithm is as follows (Sutton[3]):

- initialize $Q(s, a)$
- repeatedly do following operation
  - Select action a_t given state s_t using policy π
  - Act action a, collect reward r_t and observe successor state s_{t+1} and reward r
  - Update the Q-function using the latest experience (s_t, a_t, r_t, s_{t+1})
  - $s_t \leftarrow s_{t+1}$
- until s is final state

Given (s_t, a_t, r_t, s_{t+1}), *simple Q-learning* updates Q-value $Q(s_t, a_t)$ as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

3.2 Sensor

The equations of motion for our airship are as follows. We make two patterns. The first pattern is the pursuer has three sensors, which are horizontal distance, vertical distance (between the evader and the pursuer), and angle of the direction that the evader seen from the pursuer is. The second pattern is the pursuer has five sensors that are horizontal velocity and vertical velocity added to above three sensors. The angle sensor is divided 7 states; other sensors are divided 5 states. The number of all state in the first pattern is 175, and in the second pattern is 7000.

3.3 Action and output to propellers

We set propulsion up as follows. The unit is [N].

$$f_{gondola} = \begin{cases} 0.13 \\ 0 \\ -0.13 \end{cases} \quad f_{tail} = \begin{cases} 0.058 \\ 0 \\ -0.058 \end{cases}$$

$f_{gondola}$ is propulsion by gondola propeller, f_{tail} is propulsion by tail propeller. And we define number of action as seven states, go forward, backward, upward, downward, turn left, right, and stop.

3.4 Allocation of rewards

Reward r is allotted to the pursuer as follows.

$$r = \begin{cases} 15.0 & \text{(catch the evader)} \\ 0.03 & \text{(spot the evader 30[deg] forward)} \\ -10.0 & \text{(distance over 15[m])} \\ -10.0 & \text{(time over)} \\ -0.02 & \text{(each step and except above)} \end{cases}$$

4 Experiments

We use two airships to execute pursuit task, one is the evader and the other is the pursuer. Two airships have same dynamics that is described in Section 2.2. And the pursuer learns its action by *simple-Q-learning*. We test following five settings in order of difficulty level.

- Type1 : First distance is 5[m] and the evader does not move. First position of evader is in front of the pursuer.
- Type2 : Added Type1, the direction of pursuer to the evader is 60[deg] left.
- Type3 : Added Type2, the direction of pursuer is random among 0, 60, 120, 180, 240, and 300[deg].
- Type4 : Added Type3, the pursuer's altitude is random among 3~10[m].
- Type5 : Added Type4, the evader moves straight slowly.

The parameters of learning, $\alpha = 0.05$, $\gamma = 1.0$. The policy π is greedy policy. One trial run until the pursuer catches the evader (the horizontal distance less than 2.5[m], the vertical distance less than 1.7[m]), or the distance over 15[m], or time limit(90[s]). We repeat 15000 times in each setting. We experiment two cases. In first case, the pursuer has three sensor, two distance sensors and an angle sensor. In second case, the pursuer has five sensor, two distance sensors, an angle sensor, and two velocity sensors.

4.1 Results

Figure 4 and Figure 5 shows success rate in the first case and the second case respectively, the vertical axis represents the success rate per 100 trials, and the horizontal axis represents the trial number. The success

rate is average of five learning sets.

In the first case, the pursuer catches the evader absolutely in the type 1 and the type 2; the success rate converges on 1.0 after about 150 trials. In type2, the pursuer obtains the action turning right after going forward, and the success rate converges on 1.0 after about 250 trials. In the type 3 $\sim$ type 5, the success rate does not converge. The vertical locomotion is independent of the horizontal locomotion that is due to the characteristic of the propulsion unit; the type 3 and the type 4 are at the same level of the success rate. The type 5 is the worst, and it shows the difficulty of using only *simple-Q-learning*.

In the second case, it is the same result in respect of whether the success rate converses on 1.0. The type 1 and the type 2 need more trials to converge because the number of all states is forty times as many as the first case. It is considered that the type 3 $\sim$ the type 5 also need more trials. It might be possible to converge on 1.0 and 15000 trials are not enough.
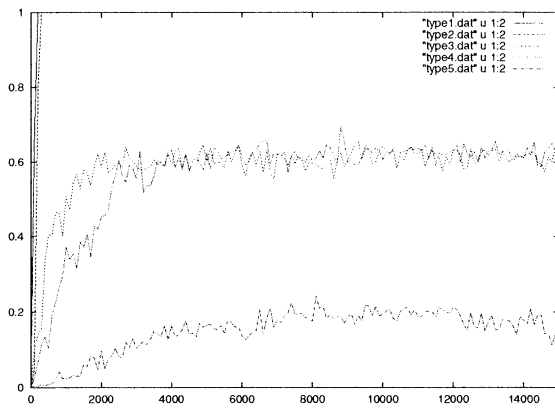
Figure 4: success rate : the first case
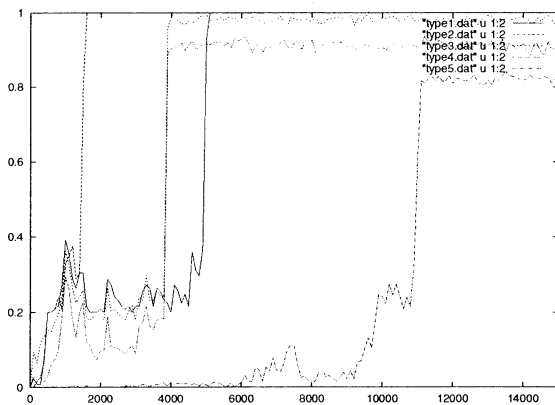
Figure 5: success rate : the second case

5 Conclusion

We apply *simple-Q-learning* to design the control system of autonomous airship and examine the possibility of learning in this paper. There is good evidence to show that *simple-Q-learning* performs well nevertheless the motion of the airship is difficult to be analyzed and in three dimension. And it performs well in the case of that is added velocity sensors than in the case of that only distance and angle sensors.

In future, we improve state configuration and allocation of rewards, or make the learner have different sensor for realization the control system that has more efficient capacity for learning. And we design the agent that can move to fixed point around itself automatically. Next, we introduce the planning agent that sets the point and design the control system that has more efficient planning mechanism. We finally develop the autonomous airship using real robot.

Acknowledgements

References

[1] Sjoerd van der Zwaan,Alexandre Bernardino,José Santos-Victor(2000). Vision based Station Keeping and Docking for an Aerial Blimp. IEEE/RSJ International Conference on Intelligent Robots and Systems -IROS'2000 - Takamatsu, Japan, October 2000

[2] Sergio B. Varella Gomes and Josue Jr. G. Ramos(1998), Airship Dynamic Modeling for Autonomous Operation, Proceedings of the 1998 IEEE International Conference on Robotics and Automation Leuven, Belgium, pp3462-3467.

[3] Richard S. Sutton and Andrew G. Barto(1998), Reinforcement Learning, MIT Press.

Remarks on Connection Methods of Neural Network Controller Using Reference Model With Conventional Controller

Takayuki Yamada

Department of Computer and Information Sciences

Faculty of Engineering

Ibaraki University
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan

Abstract

Several neural network controllers using a reference model has been proposed. One of their purposes is that we can design a conventional controller for a known linear plant through the use of neural network compensation capability for plant incorrectness such as an unknown nonlinear terms. This paper confirms that there are several connection methods for such neural network controller using a reference model and presents brief discussion with them.

1. Introduction

Many studies have been undertaken in order to apply both the flexibility and the learning capability of neural networks to control systems[1][2]. Among these studies, several neural network controllers using a reference model have been proposed in order to linearize an object plant[3][4]. When the neural network learns so as to converge a plant output to a reference model output, we can expect that the neural network learning and nonlinear mapping capabilities compensate for plant incorrectness and nonlinearity. That is, we can design a conventional linear controller for the known linear plant because the compensated plant is ostensibly linear. In this type neural network controller, it seems that there are several connection methods of the neural network with the plant, the reference model and the conventional controller. However, there is no discussion for the relationship between them and each connection characteristic.

This paper classifies the connection methods of the neural network controller using a reference model with regard to the neural network input and the plant input. It is also confirmed that there are two conventional controller designs which use the plant input-output or the reference model input-output. One type of the classified neural network controller is designed and simulated.

2. Neural network controllers using a reference model

This section outlines that the neural network controllers with a reference model are classified into D type and the Y type connections with regard to the neural network input and the plant input. A discrete time SISO (single input and single output) system is selected for the object plant. This paper does not discuss the inner structure detail of the conventional controller. This is because it does not affect the connection method of the neural network controller using a reference model in this paper discussion. A scalar conventional controller output is only discussed for the neural network design. The multi layer neural network with one output layer neuron is selected and it has no inner dynamics.

The plant and the reference model in this paper are expressed as the following equation[3].

$$Y(k)=f(Y(k-d),\cdots,Y(k-d-n),U(k-d),\cdots,U(k-d-m)) \qquad (1)$$

$$Y_M(k)=f_M(Y_M(k-d_M),\cdots,Y_M(k-d_M-n_M)$$
$$,U_M(k-d_M),\cdots,U_M(k-d_M-m_M)) \qquad (2)$$

where Y is the plant output, U is the plant input, f is the nonlinear function which expresses the plant nonlinearity, d is the dead time of the plant, n and m are the plant orders, Y_M is the reference model output, f_M is the nonlinear function which expresses the reference model nonlinearity, d_M is the dead time of the reference model, n_M and m_M are the reference model orders and k is the sampling time. The output error ε is defined as the following equation.

$$\varepsilon(k)=Y_M(k)-Y(k) \qquad (3)$$

Here, we assume that an inverse function f^{-1} of the nonlinear function f with regard to the plant input U(k-d) exists. The condition $\varepsilon(k)=0$ requires the following equation from

eqs .(1)-(3).

$$U(k-d)= f^{-1}(f_M(Y_M(k-d_M),\cdots,Y_M(k-d_M-n_M)$$
$$,U_M(k-d_M),\cdots,U_M(k-d_M-m_M))$$
$$,Y(k-d),\cdots,Y(k-d-n),U(k-d-1),\cdots,U(k-d-m)) \quad (4)$$

The following equation is obtained from eq.(4) by use of its advance to the plant dead time d future.

$$U(k)= f^{-1}(f_M(Y_M(k-d_M+d),\cdots,Y_M(k-d_M+d-n_M)$$
$$,U_M(k-d_M+d),\cdots,U_M(k-d_M+d-m_M))$$
$$,Y(k),\cdots,Y(k-n),U(k-1),\cdots,U(k-m)) \quad (5)$$

It is well known that a three layer neural network can approximate any nonlinear function in any accuracy. Thus, it can express the static nonlinear function shown in eq.(5). The following sections discuss the static nonlinear function obtained by the neural network with regard to each connection method through the use of eq.(5).

2-1 D-type

Figure 1 shows the scheme of the D-type connection. As shown here, the plant input U is the neural network output U_2, and the conventional controller output U_1 is a common input of both the neural network and the reference model. If the neural network output $U_2(k)$ becomes to be U(k) shown in eq.(5), this neural network output makes the plant output Y convergence to the reference model output Y_M. This is because the plant input is the neural network output. That is, the equation (5) is the nonlinear function which should be learned by the neural network of this type connection. As shown in eq.(5), the neural network has to learn both inverse dynamics of the plant and forward dynamics of the reference model. This fact indicates that the neural network has to learn again when the reference model is changed although the plant dynamics is not changed. The neural network input vector is expressed as the following equation from eq.(5), $U_M(k)=U_1(k)$ and $U(k)=U_2(k)$.

$$I^T(k)=[Y_M(k-d_M+d),\cdots,Y_M(k-d_M+d-n_M),$$
$$U_1(k-d_M+d),\cdots,U_1(k-d_M+d-m_M),$$
$$Y(k),\cdots,Y(k-n),U_2(k-1),\cdots,U_2(k-m)] \quad (6)$$

As shown in eq.(6), the neural network input requires future information of the reference model input (which equals the conventional controller output) when $d>d_M$. That is, we can not use this type connection if this future information is not available.

From eqs.(2) and (5), the following equation is obtained.

$$U(k)= f^{-1}(Y_M(k+d),Y(k),\cdots,Y(k-n),$$
$$U(k-1),\cdots,U(k-m)) \quad (7)$$

From eq.(7), we can also define the another neural network input vector as shown in follows:

$$I^T(k)= [Y_M(k+d),Y(k),\cdots,Y(k-n),U(k-1),\cdots,U(k-m)] \quad (8)$$

The connection using eq.(8) is called for I-type in this paper. As shown in eqs.(7) and (8), the neural network of this type connection learns the inverse dynamics of the plant and it is not required to learn the forward dynamics of the reference model. That is, the neural network size of this type is smaller than that of the D-type and the neural network has not to learn again although the reference model changed. The neural network input vector requires the dead time d future information of the reference model output.

2-2 Y-type

Figure 3 shows the scheme of Y-type. As shown in this figure, the plant input U is the sum of both the neural network output U_2 and the conventional controller output U_1. The conventional controller output U_1 is the common input of both the neural network and the reference model. From eq.(5), the following neural network output U_2 makes the convergence of the plant output Y to the reference model output Y_M.
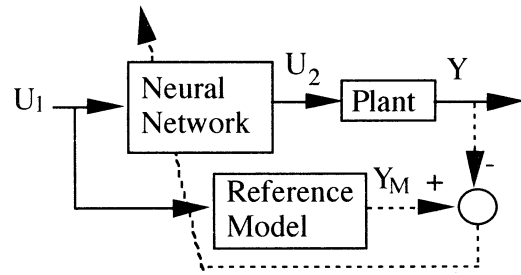
Fig.1 Scheme of D-type connection.
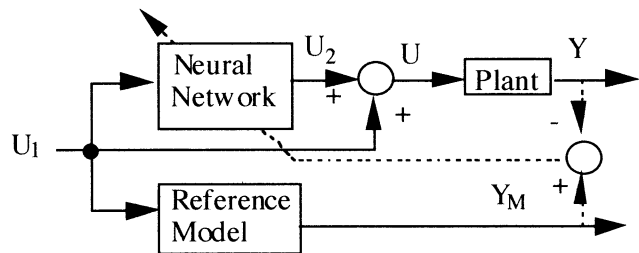
Fig.2 Scheme of Y-type connection.

$U_2(k)= f^{-1}(f_M(Y_M(k-d_M+d),\cdots,Y_M(k-d_M+d-n_M)$
$\qquad ,U_M(k-d_M+d),\cdots,U_M(k-d_M+d-m_M))$
$\qquad ,Y(k),\cdots,Y(k-n),U(k-1),\cdots,U(k-m)) - U_1(k) \quad (9)$

As shown in eq.(9), the neural network of this type connection has to learn both the inverse dynamics of the plant and the forward dynamics of the reference model. This requirement is the same as D-type. However, there is the particular requirement to learn the nonlinear function which includes the conventional controller output U_1. From eq.(9), the neural network input vector is follows:

$I^T(k)=[Y_M(k-d_M+d),\cdots,Y_M(k-d_M+d-n_M),$
$\qquad U_M(k-d_M+d),\cdots,U_M(k-d_M+d-m_M),Y(k),\cdots,Y(k-n),$
$\qquad U(k-1),\cdots,U(k-m),U_1(k)] \quad (10)$

As shown here, the neural network input vector requires the conventional controller output U_1. This fact means that the neural network size of the Y-type is larger than that of the D-type. The neural network input shown in eq.(10) requires the future information of the reference model input (which equals the conventional controller output) when $d>d_M$. This requirement is same as that of the D-type connection and if this future information is not available, we can not use the Y-type connection. When $d_M=d$, the element $U_1(k)$ of the input vector is redundant because $U_M(k)=U_1(k)$. That is, we can use the same size neural network as that of D-type.

From eqs.(2) and (9), the following equation is obtained in the same way of D-type.

$U_2(k)= f^{-1}(Y_M(k+d),Y(k),\cdots,Y(k-n),$
$\qquad U(k-1),\cdots,U(k-m)) - U_1(k) \quad (11)$

From eq.(11), the following neural network input vector is obtained.

$I^T(k)= [Y_M(k+d),Y(k),\cdots,Y(k-n),$
$\qquad U(k-1),\cdots,U(k-m),U_1(k)] \quad (12)$

When we select this neural network input vector, the neural network learns the inverse dynamics of the plant, but does not learn the forward dynamics of the reference model. That is, the neural network has not to learn again when the reference model is changed and its size is comparatively smaller. However, as shown in eq.(12), this type neural network requires the dead time d future information of the reference model.

We can design an another connection called for X-type in this paper. In this type, both the plant and the reference model have the common input which is the sum of both the

conventional output U_1 and the neural network output U_2. The neural network output U_2 is eq.(9) which makes the convergence of the plant output Y to the reference model output Y_M. However, the X-type connection has the particular feature as $U(k)=U_M(k)$ and this feature leads the comparatively smaller neural network size. We notice that the neural network learning in this type connection is stopped at some particular conditions. One example of these conditions is that $U_2(k)= -U_1(k)$ & $Y(k) =Y_M(k)$ and both the plant and the reference model have a passive characteristic.

2-3 Non separation type and separation type

This section confirms that we can classify the conventional controllers used in the above connections into non separation and separation types by using the definition of the conventional controller input. These two types can be defined for above D-type, I-type, Y-type and X-type connections.

We can define the conventional controller input as the following equations.

$U_1(k) = G_c(Y_M(k),Y_M(k-1),\cdots,U_M(k-1),U_M(k-2),\cdots) \quad (13)$
$U_1(k) = G_c(Y(k),Y(k-1),\cdots,U(k-1),U(k-2),\cdots) \quad (14)$

where G_c is the nonlinear function which expresses the nonlinearity of the conventional controller. The conventional controller shown in eq.(13) is called for the separation type in this paper. As shown here, the conventional controller input U_1 is composed of the reference model input and output. That is, we can separate both the reference model and the conventional controller from both the plant and the neural network except X-type. If necessary, the neural network can use the future information of the reference model and it mainly control the plant.

The conventional controller shown in eq.(14) is called for the non separation type in this paper. The conventional controller input U_1 is composed of the plant input and output. The neural network can not use the future information of the reference model because of this specification. That is, we can not use a part of the connections shown in sections 2-1 and 2-2. On the other hand, the non separation type has a possibility of higher robustness because the neural network has a possibility of the cooperation with the conventional controller to control the plant. Unfortunately, if this cooperation dose not perform well, this type may lead to worse results. We can design the intermediate type and it is useful in some cases.

2-4 Neural network learning rule

The neural network learns so as to minimize the squared

difference between the reference model output and the plant output. When the separation type is selected, we can easily derive the neural network learning rule by using the chain-rule except X-type[5][6]. When the non separation type or X-type is selected, its derivation is little harder, but we can obtain the neural network learning rule by using the principle of causality. The neural network learning rule includes the plant Jacobian for all types. If the inverse dynamics of the plant exists, we can overcome this problem. This is because the sign of the plant Jacobian is not changed and we can delete the plant Jacobian by including it into the parameter which determine the learning rate.

3. Simulation

To verify the above discussion, the separation D type controller for the second order discrete time plant is simulated as an example. The simulated plant and the reference model are follows:

$$Y(k) = - a_1 Y(k-1) - a_2 Y(k-2)$$
$$+ U(k-1) + bU(k-2) + C_{non}Y^2(k-1) \qquad (15)$$

$$Y_M(k) = - a_{M1}Y_M(k-1) - a_{M2}Y_M(k-2)$$
$$+ b_{M0}U_M(k-1) + b_{M1}U_M(k-2) \qquad (16)$$

Where $a_1 = -1.3$, $a_2 = 0.3$, $b = 0.7$, $a_{M1} = - 0.7$, $a_{M2} = - 0.3$, $b_{M0} = 2$, $b_{M1} = 0.2$ and $C_{non} = 0.2$ are selected. Since the separation type is selected, $U_M(k)$ equals $U_1(k)$. The proportional control is selected as the conventional controller shown in the following equation.

$$U_1(k) = Kp(Yd(k)-Y_M(k)) \qquad (17)$$

where Yd is the desired value for control and Kp is the proportional gain which 0.5 is selected for.

Figures 3 and 4 show the scheme of the block diagram and the simulated result respectively. As shown here, the simulated neural network controller using a reference model performs well.

4. Conclusion

This paper confirmed that there are several connection methods for the neural network controller using a reference model and outlined their practical design. The separation D type controller was simulated as an example and its performed well.

References

[1]K.S.Narendra and K.Parthictsarathy, "Identification and Control of Dynamics System Using Neural Networks", IEEE Transaction on Neural Networks, Vol.1, No.1, pp.4-27, 1990

[2]D.Psaltis, Siders and A.Yamamura, "Neural Network Controller", Proceedings of 1987 IEEE International Conference on Neural Networks, Vol.IV, pp.551-558, San Diego, 1987

[3]J.Tsuji, H.Ohmori and A.Sano, "Adaptive Control Incorporating Neural Network", Transactions of the Society of Instrument and Control Engineers, Vol.30, No.3, pp.295-302, 1994 (in Japanese)

[4]J.Miguchi, H.Wu, K.Mizukami, "Discrete Sliding Mode Control Using Neural Network", Transactions of the Society of Instrument and Control Engineers, Vol.33, No.8, pp.787-791, 1997 (in Japanese)

[5]Rumelhart, McClelland and the PDP Research Group, "Parallel Distributed Processing", The MIT Press, 1988

[6]S.Ohmatsu and T.Yamamoto et al., "Self Tuning Control", the Society of Instrument and Control Engineers, 1996 (in Japanese)
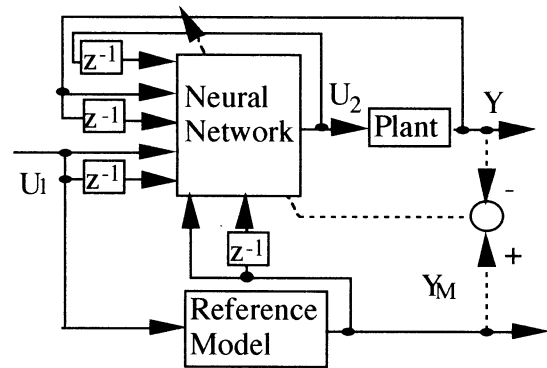
Fig.3 Block diagram of D-Type for simulation.
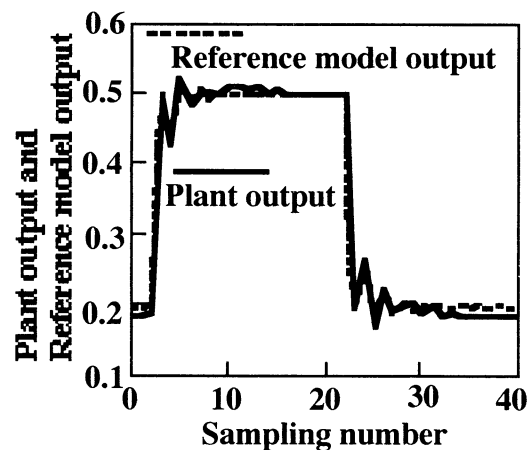
Fig.4 Simulation result.

Intelligent Control for the Vision-based Indoor Navigation

of an Alife Mobile Robot

Jiwu Wang[1]

Department of Electrical and Electronics Engineering,

Oita University, Oita 870-1192, Japan[1]

(Tel : 81-97-554-7831; Fax : 81-97-554-7841)

(Email: wang_jiwu@hotmail.com)

Hidenori Kimura[2,3]   Masanori Sugisaka[1,2]

The Institute of Physical and Chemical Research (RIKEN),

Bio-Mimetric Control Research Center, Shimoshidami,

Moriyama-ku Nagoya, 463-0003, Japan[2]

Complex Systems Department of Complexity Science and

Engineering Graduate School of Frontier Science, The

University of Tokyo, Tokyo 113-8656, Japan[3]

(Email:kimura@crux.t.u-tokyo.ac.jp, msugi@cc.oita-u.ac.jp)

Abstract

Steady and reliable navigation is a prerequisite for a mobile robot. But its reliability is often weakened by the unavoidable slip and some incorrigible drift errors for sensors, especially after a long distance navigation. Although the perceptual landmarks were solutions to such problems, it is easy to miss the landmarks at some specific spots when the robot moves at different speeds, especially at higher speeds. And if the landmarks were put in any intervals, or if the illumination conditions were not good, it will be more difficult to perceive them. In order to detect and extract the artificial landmarks robustly under different illuminating conditions, the color model of the landmarks was built in the HSV color space. The moving speed was controlled with the visual servoing control method. When the robot suddenly can not find some specific landmarks at some specific spots because of the higher speed or rapid turn of the road, it will search for the landmarks based on its intelligence, the
inertia of the previous motion, and find the landmark in the shortest time. These methods were verified by the reliable vision-based indoor navigation of an Alife mobile robot.

1. Introduction

Reliable and steady navigation is necessary for a robot moving purposefully to reach some given target locations and carrying on some specific tasks such as inspection, transportation, surveillance, or guidance etc[1]. But it can not be easily achieved because of some casual changes in environments or drift errors of sensors, etc. Using perceptual landmarks is a better method for solving such problems. Some different (color or shape) landmarks are put on some key spots, and robots use them to realize self-location and get different directions in a navigation map. Because results of image processing are serious affected by illumination conditions in environments. Although such influence can be reduced by some methods, for example, the image processing in the HSV (Hue,

Saturation and Value) color space, they are not so robust enough to get satisfactory results for all time, especially in the case of strong glistens or weaken illumination. And at this moment, the robot will have no idea about what to do in the next step.

In this paper, the visual based navigation method was applied to our Alife robot. Here the robot moves along a guideline with a different color from the floor, which determines its moving trace. Other landmarks, such as circle, triangle and cross etc, were put along the guideline to make the robot carry on different movements. It generally can move smoothly along the guideline and effectively perform specific action on the spots with the corresponding landmark at the lower speed. But when the speed was increased to some level, some landmarks will be missed, moreover sometimes the guideline will also be missed where there is a rapid turn. This is partly because there is not enough response time for image processing to recognize landmarks. The illumination conditions, is another factor that makes the landmarks not be recognized effectively. On the other hand, the robot can not move at a very slow speed because it have to overcome the friction of the ground and be appropriate to practical applications.

Therefore, the robot should have an ability to decide what to do when it can not find the given landmark, no matter it was caused by a higher speed or poor illumination conditions. In view of above problems, there should be a solution to try to help the robot find its landmark autonomously as fast as possible. Here we mainly discuss how to solve such problem with visual servo control strategy.

2. Visual servoing control for navigation

In the research of navigation of a mobile robot, the common problem is how to realize self-location after the robot moves a long distance, such as more than several hundred meters. The visual based landmark is one of the solutions. Generally, there are two categories of visual positioning system that can be used for navigation: the visual positioning and visual servoing. For the visual positioning, some research, such as the accurate camera calibration[6] that deals with intrinsic and extrinsic camera parameters, must be done first. But for the visual servoing, if the control signal can be directly calculated in the image coordinate plane, we do not need to carry on some complex camera calibration. Therefore, The visual servoing control method was used to develop navigation skills for our Alife mobile robot. Here, the visual servoing control is dependent on the geometry information of artificial landmarks in the continuous images taken by the CCD video camera. Based on the difference between the current position and the desired position of the landmarks, the robot will know its position relative to the landmarks, and then

its speed can be controlled to reach its desired position. In our experiments, a color guideline on the floor was used to mark the path along which the robot will move. At the same time, some landmarks with different shapes and colors were put along the guideline to make the robot perform some specific action and updated its present navigation map based on the present landmark position.

The position of the robot at any time may be expressed as a pair of co-ordinates (x, y) and an orientation angle (ϕ), where the coordinates represent the central point on the wheel axis.

The rate at which the position (x, y, ϕ) changes is dependent on the sum and difference of the rotational speeds of the left/right wheels (ω_L, ω_R). The speed of the left and right wheels at any time may be estimated by either counting or timing the pulses from the left and right micro-sensors which detect the slots in the encoder disks.

$$\dot{\phi} = \pi(\omega_L - \omega_R)d / D$$

$$\dot{x} = 0.5(\omega_L + \omega_R)\pi d \sin(\phi)$$

$$\dot{y} = 0.5(\omega_L + \omega_R)\pi d \cos(\phi)$$

where d represents the wheel diameter and D represent the wheel distance.

Now, the visual servo control for our mobile robot can be summary in Figure 1:
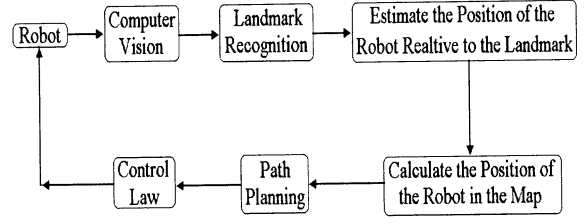
Figure 1 The illustration of visual servo control in the Alife mobile robot

It is not necessary to get the accurate position of landmarks for a visual servo control system to control the navigation. This makes the image processing easier. But it also brings new problem. That means the robot has to adjust its speed in time. Otherwise, it will miss the landmark suddenly, especially when there is a rapid turn or occasional poor lighting.

We all usually have the experience to drive a car at a clear noon in summer. Sometimes we can not see parts of the white central line of the highway at ahead of the car because of the glistens. Then we will adjust our angle of view or our driving speed till we can see road sign clearly. This method can be transplanted to our robot. But unlike human beings with high intelligence, who can have various solutions to most casual cases, the robot should have a simple solution.

As we know, the robot should always keep the last successful position when it gets a new position. If the next new position can not find, such as before a rapid turn or landmarks with glisten, the robot will stop and turn a few degrees based on the inertia of previous position corresponding to the

central line of its eyeshot. The next problem is if the kept position is in the fastest searching direction. In the case of a rapid turn, mostly that searching method is the fastest solution. While for glistens, the searching direction was not so important because it will usually find the landmarks after a few angle's adjustment. If it is the combination of a rapid turn and glisten, the probability is equal for the right and wrong direction. With this method, the robot will find its next new position as fast as possible when the landmark suddenly disappears. At the same time, it is also helpful to sheer off the strong reflections from the landmarks or the floor and finally recognize the specific landmark accurately. The illustration was given in figure 1.
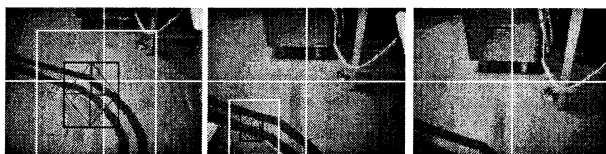
Figure 2 The illustration of a rapid turn

3. Conclusions

Perceptual landmarks are ·effective solutions to the slip or unavoidable drift errors of sensors in a long distance navigation for a visual based mobile robot. In order to find landmarks effectively, we make the robot to search landmark based on the inertia of previous position. This strategy was verified by indoor navigation experiments.

References

[1] J. Wang, M. Sugisaka, Study on visual-based indoor navigation for an alife mobile robot, ICSEng2002, Las Vegas, USA, 2002, August 6-8 : 390-396.

[2] Masanori Sugisaka, Xin Wang, Ju-Jung Lee, Intelligent control with new image processing strategy for a mobile vehicle, Artificial life robotics, 1998: 113-118.

[3] J. Wang, M. Sugisaka, Study on a color based target detection technique", Proceedings of 7-th international symposium in artificial life and robotics (AROB 7-th'02), B-Con Plaza, Beppu, Oita, Japan, Jan. 16-18 : 532-536

[4] J. Wang, M. Sugisaka, Improvement on the image processing for an automous mobile robot with an intelligent control system, International conference on control, automation and system (ICCAS2001), 2001: 437-440

[5] J. Wang, M. Sugisaka, Improve digit recognition capability of backpropagation neural networks by enhancing image preprocessing technique, International conference on control, automation and system (ICCAS2001), 2001 : 280-283

[6] J. Wang, M. Sugisaka, Study on calibration of stereo cameras for a mobile robot prototype, Proceedings of 7-th international symposium in artificial life and robotics (AROB 7-th'02), B-Con Plaza, Beppu, Oita, Japan, Jan. 16-18 : 259-262

Robust Decentralized Control and Robust Output Tracking for a Class of Linear Uncertain Interconnected Systems with Unmatched Interconnections and Uncertainties

Zheng Wang

Shenyang Institute of Automation,
Chinese Academy of Sciences,
Shenayng, 110016, P.R.China
(Email: mousewz@sia.ac.cn)

Xiongfeng Feng, Masanori Sugisaka[1,2]

[1]Department of Electrical and Electronics
Engineering, Oita University, Oita 870-1192, Japan
(Email: fxf@cc.oita-u.ac.jp,
msugi@cc.oita-u.ac.jp)

Hidenori Kimura

[2]The Institute of Physical and Chemical
Research (RIKEN), Bio-Mimetric Control
Research Center, Shimoshidami,
Moriyamaku, Nagoya, 463-0003, Japan
Department of Complexity Science and
Engineering Graduate School of Frontier
Science, The University of Tokyo, Tokyo
113-8656, Japan
(Email:kimura@crux.t.u-tokyo.ac.jp)

Abstract

In this paper, we present a robust decentralized control and output tracking method for a class of linear uncertain interconnected systems with unmatched interconnections and uncertainties. By using two-step method, we take full advantage of system interconnections and avoid their negative effect in the design of robust decentralized controller. Unmatched uncertainties are also considered. By using similar technique, the design of output tracking controllers is given. The illustrative examples are given to show the efficiency of our method.

Keywords: interconnection, unmatched, robust control, decentralized control, output tracking.

1 Introduction

The decentralized control schemes for large-scale interconnected systems have become more interesting in the recent years. One of the benefits with decentralized control is that the interconnected system can be decomposed into several lower order subsystems, therefore the design and implementation of each subsystem can proceed independently. The main objectives of decentralized control schemes dealing with linear or nonlinear interconnected systems are to find some feedback laws for adapting the interactions from the other subsystems where no state information is transferred. But the interconnections ([1-4]) were dealt with as disturbance in these methods. In fact, the interconnections act a key role in the composite systems. The advantage of interconnections should be used to design decentralized controllers, or it will increase the conservatism of systems.

Tracking is a general practical and important problem in control system design, and it is one of the problems difficult to solve. Because of the uncertainties in the system model, the problem of robust output tracking has been given much attention. In [5-7] and [8-9], the robust output tracking control are studied respectively for linear systems and nonlinear systems with uncertainties.

In this paper, we present a decentralized robust tracking method for a class of linear uncertain interconnected systems with unmatched interconnections and uncertainties. By using two-step method, the advantages of interconnections can be utilized in a better way and unmatched uncertainties are taken into consideration, which usually cannot be studied in the problem of robust decentralized control and robust output tracking. By using similar technique, the design of output tracking controllers is put forward. Illustrative

examples are given to show the efficiency of the proposed methods.

2 System Description

Consider the following linear uncertain interconnected system, which is an interconnection of i subsystems described by

$$\dot{x}_i = (\mathbf{A}_i + \Delta\mathbf{A}_i)x_i + \sum_{i=1}^{N}(\mathbf{H}_{ij} + \Delta\mathbf{H}_{ij})x_j + (\mathbf{B}_i + \Delta\mathbf{B}_i)u_i$$
$$i = 1,2,\cdots,N. \tag{1}$$

1. $x_i \in R^n$ and $u_i \in R^m$ are the state and control vectors respectively.

2. $\mathbf{A}_i$ and $\mathbf{B}_i$ are with appropriate dimensions representing the system and input matrices, respectively.

3. $\Delta\mathbf{A}_i$ and $\Delta\mathbf{B}_i$ are the system and input uncertainties with the same dimensions as $\mathbf{A}_i$ and $\mathbf{B}_i$, respectively.

4. $\mathbf{H}_{ij}$ is the interconnection matrix imposed on he i th subsystem from the j th subsystem with an appropriate dimension and $\mathbf{H}_{ii} = 0$.

5. $\Delta\mathbf{H}_{ij}$ is the interconnection uncertainty with the same dimension as $\mathbf{H}_{ij}$.

The compact form of system (1) is as follows.

$$\dot{x} = (\mathbf{A} + \Delta\mathbf{A})x + (\mathbf{B} + \Delta\mathbf{B})u \tag{2}$$

where $x = [x_1^T, x_2^T, \cdots, x_N^T]^T$, $u = [u_1^T, u_2^T, \cdots, u_N^T]^T$,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{H}_{12} & \mathbf{H}_{13} & \cdots & \mathbf{H}_{1N} \\ \mathbf{H}_{21} & \mathbf{A}_2 & \mathbf{H}_{23} & \cdots & \mathbf{H}_{2N} \\ \mathbf{H}_{31} & \mathbf{H}_{32} & \mathbf{A}_3 & \cdots & \mathbf{H}_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{N1} & \mathbf{H}_{N2} & \mathbf{H}_{N3} & \cdots & \mathbf{A}_N \end{bmatrix},$$

$$\Delta A = \begin{bmatrix} \Delta\mathbf{A}_1 & \Delta\mathbf{H}_{12} & \Delta\mathbf{H}_{13} & \cdots & \Delta\mathbf{H}_{1N} \\ \Delta\mathbf{H}_{21} & \Delta\mathbf{A}_2 & \Delta\mathbf{H}_{23} & \cdots & \Delta\mathbf{H}_{2N} \\ \Delta\mathbf{H}_{31} & \Delta\mathbf{H}_{32} & \Delta\mathbf{A}_3 & \cdots & \Delta\mathbf{H}_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \Delta\mathbf{H}_{N1} & \Delta\mathbf{H}_{N2} & \Delta\mathbf{H}_{N3} & \cdots & \Delta\mathbf{A}_N \end{bmatrix},$$

$\mathbf{B} = diag\{\mathbf{B}_1, \mathbf{B}_2, \cdots, \mathbf{B}_N\}$,

$\Delta\mathbf{B} = \{\Delta\mathbf{B}_1, \Delta\mathbf{B}_2, \cdots, \Delta\mathbf{B}_N\}$.

In addition to the above descriptions of our problem, the following assumptions are also made.

A1. There exist matrices with appropriate dimensions $\overline{\mathbf{U}}_{ij}$, $\overline{\mathbf{V}}_{ij}$, $\widetilde{\mathbf{U}}_i$ and $\widetilde{\mathbf{V}}_i$ such that

$$\Delta\mathbf{A}_{ij} = \overline{\mathbf{Y}}_{ij}\mathbf{F}(\cdot)\overline{\mathbf{V}}_{ij} \quad \Delta\mathbf{B}_i = \widetilde{\mathbf{U}}_i\mathbf{F}(\cdot)\widetilde{\mathbf{V}}_i,$$

and

$$\mathbf{F}(\cdot) \in \Phi \overset{\Delta}{=} \{\mathbf{F}(t) | \mathbf{F}^T(t)\mathbf{F}(t) \leq \mathbf{I}, \quad \forall t \in R\},$$

where Φ is a compact set.

A2. $Rank\begin{bmatrix} \mathbf{A}_i & \mathbf{B}_i \\ \mathbf{C}_i & \mathbf{0} \end{bmatrix} = n_i + l_i$

3 The Design of Robust Decentralized Controllers

The interconnections of considered system (1) are known. In order to design the decentralized controller based on the advantages of interconnections, a two-step approach is proposed.

Firstly, find the solution $\mathbf{K}$, which makes $\mathbf{A} + \mathbf{BKA}_0$ Volterra-Lyapunov stable.

Here $\mathbf{A}_0 = diag\{\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_N\}$.

Usually, it is difficult to get the solution $\mathbf{K}$ by Riccati equation or inequality. LMI approach is used here. Its superiority is that the different parameters can be solved optimally at the same time.

Then, by using $\mathbf{K}$, the new form of the system (2) can be got whose interconnections satisfy match condition.

In order to get robust decentralized controller, the uncertainties are considered.

$\mathbf{X} = diag\{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_N\}$, $\mathbf{Y}_1$, $\mathbf{Y}_2$ and $\mathbf{Q}$ are parameters to be determined, $\mathbf{Y}_2$ is a diagonal block matrix and has the same form as $\mathbf{X}$. $\mathbf{Q}$ is a positive-definite matrix.

If the positive-definite solution exists in the following LMIs

$$\begin{bmatrix} \mathbf{XA}^T + \mathbf{AX} + \mathbf{BY}_1 + \mathbf{Y}_1^T \mathbf{B}^T + \varepsilon_1 \overline{\mathbf{U}}\overline{\mathbf{U}}^T + \mathbf{Q} & \mathbf{X}\overline{\mathbf{V}}^T \\ \overline{\mathbf{V}}\mathbf{X} & -\varepsilon_1 \mathbf{I} \end{bmatrix} < 0 \quad (3)$$

let $\mathbf{K}_1 = \mathbf{Y}_1 \mathbf{X}^{-1}$, using $\mathbf{K}_1$ got from LMI (3), system (2) can be changed into the following equivalent form

$$\dot{x} = (\hat{\mathbf{A}} + \Delta\mathbf{A})x + \hat{\mathbf{H}}x + (\mathbf{B} + \Delta\mathbf{B})u \quad (4)$$

where

$$\hat{\mathbf{A}} = \mathbf{A} + \mathbf{BK} \quad \Delta\mathbf{A} = \overline{\mathbf{U}}\mathbf{F}(\cdot)\overline{\mathbf{V}},$$

$$\Delta\mathbf{B} = \tilde{\mathbf{U}}\mathbf{F}(\cdot)\tilde{\mathbf{V}}, \quad \hat{\mathbf{H}} = -\mathbf{BK},$$

$$\overline{\mathbf{U}} = \operatorname*{diag}_{i=1,\cdots,N}\{\overline{\mathbf{U}}_{i1} \quad \cdots \quad \overline{\mathbf{U}}_{iN}\}$$

$$\overline{\mathbf{V}} = \left[\operatorname*{diag}_{j=1,\cdots,N}\{\overline{\mathbf{V}}_{1j}\}^T \quad \cdots \quad \operatorname*{diag}_{j=1,\cdots,N}\{\overline{\mathbf{V}}_{Nj}\}^T \right]^T$$

$$\tilde{\mathbf{U}} = \operatorname*{diag}_{i=1,\cdots,N}\{\tilde{\mathbf{U}}_i \quad \cdots \quad \overline{\mathbf{U}}_i\}$$

$$\tilde{\mathbf{V}} = \left[\operatorname*{diag}_{j=1,\cdots,N}\{\tilde{\mathbf{V}}_1^T\} \quad \cdots \quad \operatorname*{diag}_{j=1,\cdots,N}\{\tilde{\mathbf{V}}_N^T\} \right]^T$$

Then, computing the following LMI

$$\begin{bmatrix} -\mathbf{Q} + \mathbf{B}(\mathbf{Y}_2 - \mathbf{Y}_1) + (\mathbf{Y}_2 - \mathbf{Y}_1)^T \mathbf{B}^T + \varepsilon_2 \tilde{\mathbf{U}}\tilde{\mathbf{U}}^T & \mathbf{Y}_2^T \tilde{\mathbf{V}}^T \\ \tilde{\mathbf{V}}\mathbf{Y}_2 & -\varepsilon_2 \mathbf{I} \end{bmatrix} < 0 \quad (5)$$

Set $\mathbf{K}_2 = \mathbf{Y}_2 \mathbf{X}^{-1}$. Notice $\mathbf{K}_2$ is diagonal block matrix too. If $\mathbf{K}_2 = diag\{\mathbf{K}_2^1, \mathbf{K}_2^2, \cdots, \mathbf{K}_2^N\}$, then $\mathbf{K}_2^i = \mathbf{Y}_2^i \mathbf{X}_i^{-1}$.

The decentralized controller is designed by

$$u_i = \mathbf{K}_2^i x_i \quad (6)$$

Theorem 1: Under the assumption A1, if there exist some positive value ε_1 and ε_2, LMI (3) has the positive-definite solution and LMI (4) has solution, then the closed system composed of system (2) and decentralized controller (6) is asymptotically stable.

Proof. Lyapunov function can be constructed as

$$V(x,t) = x^T \mathbf{P}x,$$

where $\mathbf{P}$ is the inverse of the positive-definite solution $\mathbf{X}$ of LMI (3).

When using assumption A1, we have

$$\dot{V}(x,t) = 2x^T(t)\mathbf{P}\dot{x}(t)$$

$$= 2x^T \mathbf{P}((\hat{\mathbf{A}} + \Delta\mathbf{A})x + (\mathbf{B} + \Delta\mathbf{B})u + \hat{\mathbf{H}}x)$$

$$= x^T\left(\mathbf{P}\hat{\mathbf{A}} + \hat{\mathbf{A}}^T\mathbf{P} + \varepsilon_1 \mathbf{P}\overline{\mathbf{U}}\mathbf{F}(\cdot)\mathbf{F}(\cdot)^T\overline{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_1}\overline{\mathbf{V}}^T\overline{\mathbf{V}}\right)$$

$$+ x^T(\mathbf{P}(\hat{\mathbf{H}} + \mathbf{BK}_2) + (\hat{\mathbf{H}} + \mathbf{BK}_2)^T\mathbf{P})x$$

$$+ x^T\left(\varepsilon_2 \mathbf{P}\tilde{\mathbf{U}}\mathbf{F}(\cdot)\mathbf{F}(\cdot)^T\tilde{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_2}\mathbf{K}_2^T\tilde{\mathbf{V}}^T\tilde{\mathbf{V}}\mathbf{K}_2\right)x$$

$$\leq x^T(\mathbf{P}(\mathbf{A} + \mathbf{BK}_1) + (\mathbf{A} + \mathbf{BK}_1)^T\mathbf{P})x$$

$$+ x^T\left(\varepsilon_1 \mathbf{P}\overline{\mathbf{U}}\overline{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_1}\overline{\mathbf{V}}^T\overline{\mathbf{V}}\right)x$$

$$+ x^T(\mathbf{P}(\mathbf{BK}_2 - \mathbf{BK}_1) + (\mathbf{BK}_2 - \mathbf{BK}_1)^T\mathbf{P})x$$

$$+ x^T\left(\varepsilon_2 \mathbf{P}\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_2}\mathbf{K}_2^T\tilde{\mathbf{V}}^T\tilde{\mathbf{V}}\mathbf{K}_2\right)x$$

Since LMI (3) has solutions, that is to say,

$$\mathbf{P}(\mathbf{A} + \mathbf{BK}_1) + (\mathbf{A} + \mathbf{BK}_1)^T\mathbf{P} + \varepsilon_1 \mathbf{P}\overline{\mathbf{U}}\overline{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_1}\overline{\mathbf{V}}^T\overline{\mathbf{V}}$$

$$+ \mathbf{P}(\mathbf{BK}_1 - \mathbf{BK}_2) + (\mathbf{BK}_1 - \mathbf{BK}_2)^T\mathbf{P}$$

$$+ \varepsilon_2 \mathbf{P}\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T\mathbf{P} + \frac{1}{\varepsilon_2}\mathbf{K}_2^T\tilde{\mathbf{V}}^T\tilde{\mathbf{V}}\mathbf{K}_2 < 0$$

We can get,

$$\dot{V}(x,t) < 0$$

4 Output Tracking

Similar with [10], the problem of output tracking is studied. Interconnections and interconnection uncertainties are considered. Moreover, the interconnections do not satisfy the matching condition. The system is described by

$$\dot{x}_i = (\mathbf{A}_i + \Delta\mathbf{A}_i)x_i + \sum_{i=1}^N (\mathbf{H}_{ij} + \Delta\mathbf{H}_{ij})x_j + (\mathbf{B}_i + \Delta\mathbf{B}_i)u_i + d_i$$

$$y_i = \mathbf{C}_i x_i, \quad i = 1, 2, \cdots, N. \quad (7)$$

where $y_i \in R^{l_i}$ is output vector. $d_i \in R^{n_i}$ is the constant but unknown disturbance associated with the i th subsystem.

From upper description of the problem, we can constructed the augmented system equations as

$$\dot{x}_i = (A_i + \Delta A_i)x_i + \sum_{i=1}^{N}(H_{ij} + \Delta H_{ij})x_j + (B_i + \Delta B_i)u_i + d_i$$

$$\dot{q}_i(t) = C_i x_i(t) - y_{ri} \ , \ i = 1,2,\cdots,N. \quad (8)$$

where $q_i(t) \in R^{l_i}$ is an auxiliary state and $y_{ri} \in R^{l_i}$ is the constant reference output vector.

Rewriting (8) into an augmented state equation, we have

$$\dot{z}_i(t) = (A_{zi} + \Delta A_{zi})z_i(t) + \sum_{\substack{j=1 \\ j\neq i}}^{N}(H_{zij} + \Delta H_{zij})z_j$$

$$+ (B_{zi} + \Delta B_{zi})u_i + d_{zi} \quad (9)$$

where

$$z_i = \begin{bmatrix} x_i(t) \\ q_i(t) \end{bmatrix}, \quad A_{zi} = \begin{bmatrix} A_i & 0 \\ C_i & 0 \end{bmatrix}, \quad \Delta A_{zi} = \begin{bmatrix} \Delta A_i & 0 \\ 0 & 0 \end{bmatrix},$$

$$B_{zi} = \begin{bmatrix} B_i \\ 0 \end{bmatrix}, \quad \Delta B_{zi} = \begin{bmatrix} \Delta B_i \\ 0 \end{bmatrix}, \quad H_{zij} = \begin{bmatrix} H_{ij} & 0 \\ 0 & 0 \end{bmatrix},$$

$$\Delta H_{zij} = \begin{bmatrix} \Delta H_{ij} & 0 \\ 0 & 0 \end{bmatrix}, \quad d_{zi} = \begin{bmatrix} d_i \\ y_{ri} \end{bmatrix}.$$

Writing system (9) into compact form

$$\dot{z} = (A_z + \Delta A_z)z + (B_z + \Delta B_z)u + d_z \quad (10)$$

where $d_z = [d_{z1}^T, d_{z2}^T, \cdots, d_{zN}^T]^T$. The definitions of z , A_z , ΔA_z , B_z and ΔB_z are similar to those in equation (2).

It is not difficult to get the fact that ΔA_z and ΔB_z are still satisfy the assumption A1. In this case, take

$$\overline{U}_{zij} = \begin{bmatrix} U_{ij} \\ 0 \end{bmatrix} \ \overline{V}_{zij} = \begin{bmatrix} V_{ij} & 0 \end{bmatrix} \ \tilde{U}_{zij} = \begin{bmatrix} \tilde{U}_{ij} \\ 0 \end{bmatrix}, \ \tilde{V}_{zij} = \tilde{V}_{ij}$$

Theorem 2: Under assumptions A1 and A2, if there exist positive value ε_1 and ε_2, and the following LMI

$$\begin{bmatrix} X_z A_z^T + A_z X_z + B_z Y_{1z} + Y_{z1}^T B_z^T + \varepsilon_1 \overline{U}_z \overline{U}_z^T + Q_z & X_z \overline{V}_z^T \\ \overline{V}_z X_z & -\varepsilon_1 I \end{bmatrix} < 0 \quad (11)$$

$$\begin{bmatrix} -Q_z + B_z(Y_{z2} - Y_{z1}) + (Y_{z2} - Y_{z1})^T B_z^T + \varepsilon_2 \tilde{U}_z \tilde{U}_z^T & Y_{z2} \tilde{V}_z^T \\ \tilde{V}_z Y_{z2}^T & -\varepsilon_2 I \end{bmatrix} < 0 \quad (12)$$

(where $X_z = diag\{X_{z1}, X_{z2}, \cdots, X_{zN}\}$, Y_{z1} , Y_{z2} and Q_z are parameters to be determined, where Y_{z2} is a diagonal block matrix and has the same form as X_z . Q_z is a positive-definite matrix.) has solutions, then the system (7) can track asymptotically the given reference output, and the robust decentralized tracking controller is

$$u_i(t) = K_{z2}^i z_i(t) = Y_{z2}^i X_{zi}^{-1} z_i(t) \quad (13)$$

Proof.

(i) Internal stability

When d_i is first neglected in the augmented system (9), the form is same as that of system (1). According to Theorem 1, when the system is controlled by (13), it is asymptotically stable.

(ii) Asymptotic tracking

When the feedback gain for each subsystem is chosen as K_{z2}^i, then the global closed-loop system equation will be

$$\dot{z}_i(t) = (A_z + \Delta A_z + (B_z + \Delta B_z)K_{z2})z(t) + d_z$$

$$\equiv A_c z(t) + d_z \quad (14)$$

where K_{z2} is the block-diagonal matrix of K_{z2}^i, for $i = 1,2,\cdots,N$.

Differentiating both sides of (14), we have

$$\ddot{z}(t) = A_c \dot{z}(t) \quad (15)$$

Since A_c is a strictly stable matrix from (i), $\dot{z}(t)$ will approach 0 no matter what the initial condition is, that is,

$$\dot{q}(t) = y_i(t) - y_r \to 0 \quad (16)$$

as $t \to \infty$ for all $i = 1,2,\cdots,N$.

From (16), it is obvious that the property of asymptotic tracking is achieved. Combining (i) and (ii), the proof is completed.

5　Illustrative Examples

Example 1: To illustrate the design of Theorem 1, we consider the following linear uncertain interconnected

system composed of two two-dimensional subsystems described by nominal system:

$$A = \begin{bmatrix} 0 & 1 & & \\ 1 & 2 & & \\ & & 0 & 1 \\ & & 1 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \quad H = \begin{bmatrix} & & 1 & 2 \\ & & 2 & 1 \\ 1 & 2 & & \\ 3 & 1 & & \end{bmatrix}$$

Uncertainty matrices

$$\Delta A_i = \Delta H_{ij} = \begin{bmatrix} 0 & \delta \\ 0 & 0 \end{bmatrix}, \quad |\delta| < 0.05$$

From the above descriptions, it is obvious that the assumption A1 is satisfied.

In the environment of LMITOOL, we can get the solutions of LMI (10),

$$Y_1 = \begin{bmatrix} -2.8 & -17.3 & -22.0 & -1180.4 \\ -31.8 & -1192.4 & -8 & -17.3 \end{bmatrix}$$

$$Y_2 = \begin{bmatrix} -2.8 & -222.8996 & & \\ & & -8 & -222.8996 \end{bmatrix}$$

$$K_2 = \begin{bmatrix} -30.4912 & -61.5910 & & \\ & & -31.3660 & -62.2904 \end{bmatrix}$$

According to Theorem 1, the decentralized controller is designed as

$$u_1 = -30.4912x_{11} - 61.2910x_{12}$$

$$u_2 = -31.3660x_{21} - 62.2904x_{22}$$

Example 2: To illustrate the design of Theorem 2, we consider the linear uncertain interconnected system composed of two two-dimensional subsystems described by example 1 again.

Output matrix:

$$C_i = \begin{bmatrix} 1 & -1 \end{bmatrix}^T$$

Reference output vector:

$$y_r = \begin{bmatrix} 1 & 2 \end{bmatrix}^T$$

Disturbance vector:

$$d = \begin{bmatrix} 2 & 1 & 1 & 2 \end{bmatrix}^T$$

In the environment of LMITOOL, we can get the solutions of LMI (10),

$$Y_1 = \begin{bmatrix} -164.8306 & -338.1282 & 24.4641 & -321.3855 & 55.1704 & -49.3683 \\ -376.6445 & -261.0334 & -59.5583 & -164.8306 & -338.1282 & 24.4641 \end{bmatrix}$$

$$Y_2 = \begin{bmatrix} -170 & -88860 & 20 & & & \\ & & & -40 & -117280 & 40 \end{bmatrix}$$

$$K_2 = \begin{bmatrix} -39.5 & -878.6 & 114.9 & & & \\ & & & -48.7 & -1159.6 & 151.5 \end{bmatrix}$$

Fig.1-4 gives state response of subsystem 1 and 2 in Example 1, output response of subsystem 1 and 2 in Example 1, respectively.
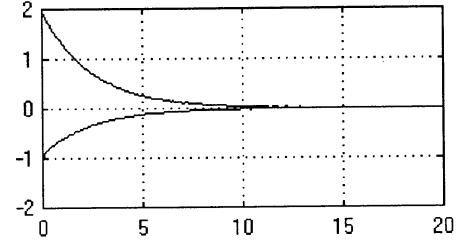
Fig. 1 State response of subsystem 1 in Example 1
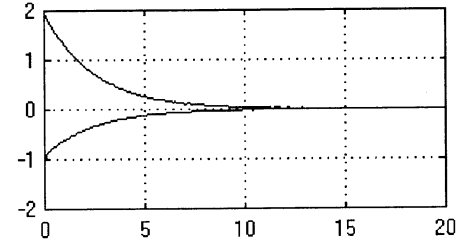
Fig. 2 state response of subsystem 2 in Example 1
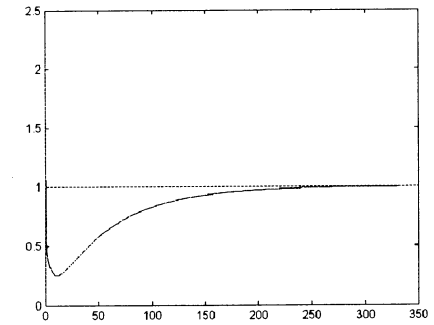
Fig. 3 Output response of subsystem 1 in Example 1
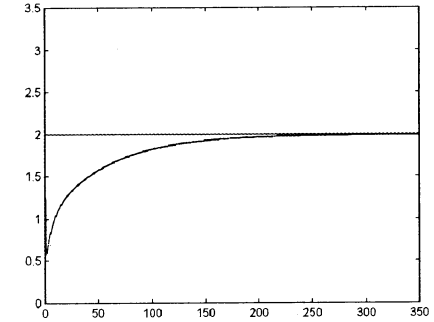
Fig. 4 Output response of subsystem 2 in Example 1

6 Conclusion

A robust decentralized control and output tracking method for a class of linear uncertain interconnected systems with unmatched interconnections and uncertainties is proposed. Related theorems and proofs are given. By using two-step method, full advantage of system interconnections is made use of and their negative effect is avoided in the design of robust decentralized controller, with considering of unmatched uncertainties. Similar technique is used in the design of output tracking controllers. Illustrative examples show the efficiency of the proposed method.

Reference

[1] Hassan, M. F. and M. G. Singh (1980). Decentralized controller with online interaction improvement. IEEE Proc. D, 127, 142-148.

[2] Ikeda, M. and D. D. Wiljak (1980). Decentralized stabilization of linear time-varying systems. IEEE Trans. Autom. Control, 25, 106-107.

[3] Mao, C, J. and W. S. Lin (1990). Decentralized control of interconnected systems with unmodelled nonlinearity and interaction. Automatica, 26, 263-268.

[4] Sakscna, V. R, and J. B. Cruz (1985). An approach to decentralized control of large-scale systems using aggregation methods. IEEE Trans. Autom. Control, 30, 912-914.

[5] Liu X. P., Output regulation of strongly coupled symmetric composite systems, 1992, 28(4): 1037-1041.

[6] Schmitendorf W. E. and B. R. Barmish, Robust asymptotic tracking for linear systems with unknown parameters, Automatica, 1986, 22(2); 335-360.

[7] Schmitendorf W. E., Methods for obtaining robust tracking control laws, Automatica, 1987, 23(2): 675-677.

[8] Betash S., Robust output tracking for control of nonlinear systems, Int. J. Contr., 1990, 21: 1381-1407.

[9] Hakan E. and N. Olgac, Robust output tracking control of nonlinear MIMO systems via sliding mode technique, Automatica, 1992, 28(1): 145-151.

[10] Mao C. J. and J. H. Yang, Decentralized stabilization and output tracking of large-scale systems, 1996, 32(7): 10077-1080.

Situated and Embodied Evolution in Collective Evolutionary Robotics

Yukiya USUI   Takaya ARITA

Graduate School of Human Informatics, Nagoya University

Furo-cho, Chikusa-ku, Nagoya 464-8601, Japan

E-mail :{yukiya@create, ari@info}.human.nagoya-u.ac.jp

Abstract

Evolutionary robotics is a challenging technique for creation of autonomous robots based on the mechanism of Darwinian evolution. In the conventional evolutionary robotics, the "simulate-and-transfer" method has been adopted. We believe that the most likely candidate methodology in evolutionary robotics for near future is "Situated and Embodied Evolution", in which real robots in real world evolve based on the interactions with actual environment and real robots. It becomes important when realizing Situated and Embodied Evolution to decentralize the algorithm for evolution computation, because it could make implementation of efficient systems easier and could accelerate diversification in robot behavior. This paper proposes a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems that realize Situated and Embodied Evolution. This paper also reports on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented.

Keywords: Situated and embodied evolution, Evolutionary robotics, Genetic algorithm.

1   Introduction

Evolutionary robotics is a challenging technique for creation of autonomous robots based on the mechanism of Darwinian evolution [1]. In the conventional evolutionary robotics, the "simulate-and-transfer" method has been adopted (Figure 1(a)). However, several issues are increasingly problematic for the method.

1) It is very difficult or takes long time to simulate complex behavior of robots and complex environment.

2) It is necessary to model the environment every time when a new task is given.

3) Scalability to the number of the robots is poor in case of the systems with a population of robots having
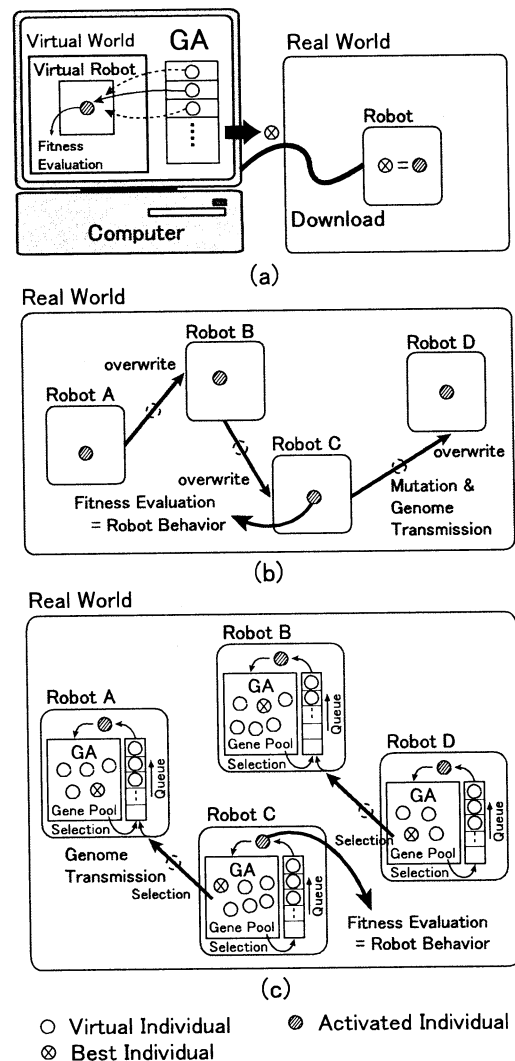
Figure 1: Schematic diagram for evolution of robots: (a) "Simulate-and-transfer" method, (b) "1 genome per robot" method, (c) Proposed method.

complex interaction among them.

We believe that the most likely candidate methodology in evolutionary robotics for near future is "Situated and Embodied Evolution", in which real robots in real world evolve based on the interactions with actual environment and real robots. It becomes important when realizing Situated and Embodied Evolution to decentralize the algorithm of evolution (genetic algorithm), because decentralization of evolutionary computation could make implementation of efficient systems easier and could accelerate diversification in robot behavior.

Very few studies regarding Situated and Embodied Evolution have been conducted. Among them, Watson et al. proposed a method to realize Situated and Embodied Evolution based on a motivation that is similar to ours described above [2]. They adopted a straightforward method for evolutionary computation, in which each robot represents one individual and population share their genetic information by transmitting information among them when they encounter (Figure 1(b)). However, the progress of evolution in this method directly depends on the number of the robots and the frequency of encounter with other robots.

This paper proposes a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems with adequate scalability that realize Situated and Embodied Evolution. There are two levels of optimization in this method (Figure 1(c)). There is transmission of good genes among robots when they encounter. Also, each robot executes a genetic algorithm within itself by emulating many "virtual individuals" based on time-sharing. This method thus reduces dependence of the number of the robots and of the frequency of encounter with other robots on the speed of evolution, which can realize flexible and efficient robotic systems with adequate scalability. This paper also reports on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented.

2 A Model for Situated and Embodied Evolution

Parallelization of genetic algorithms (GA) has been discussed in the field of evolutionary computation, motivated mainly by the desire to reduce the overall computation time. Most of the proposed parallel GAs fall into a class which has come to be called "island model" parallel GA. Island model parallel GA divides a population into subpopulations and assigns them to processing elements on a parallel or distributed computer. Then each subpopulation searches the optimal solution independently, and exchanges individuals periodically.

Our distributed genetic algorithm for Situated and Embodied Evolution can be called island model parallel GA in that each robot has a subpopulation, searches the optimal solution, and exchange good individuals. However, there is a significant difference in our model from the conventional island model parallel GA as follows.

1) Communication topology and frequency are dynamic, which depends on the robot behavior, especially encounters of robots.

2) Fitness evaluation is conducted as robot behavior in real word, which needs quite long time compared with other evolutionary operations done in the robots.

3) Optimal solution varies depending on the behavior range and physical characteristics of each robot, besides the dynamic property of the environments.
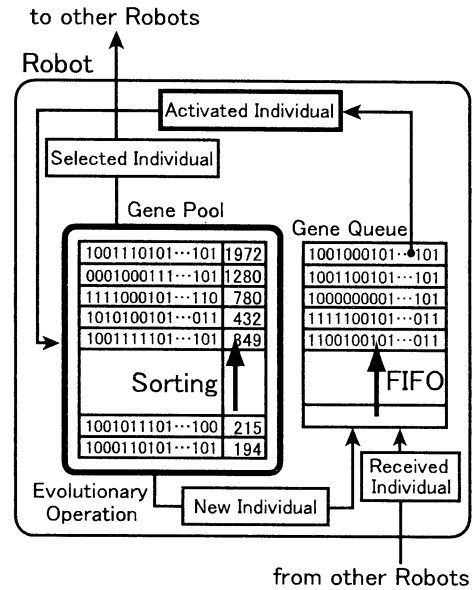
Figure 2: Evolutionary processing in each robot.

Typical implementation of evolutionary computation in each robot is shown in Figure 2. Each robot has a "gene pool" and a "gene queue". The gene pool has an evolved subpopulation, whose individuals are expressed as genomes and sorted by their fitness values. A new individual is generated by selecting (copying) 2 individuals from the gene pool based on roulette wheel selection, and operating one point crossover and mu-

tation. The new individual is then put into the gene queue, and waits for being evaluated. New individuals migrated from other robots are also put into the gene queue, and reevaluated in this robot, because there can be difference in their environments and physical characteristics among robots. A dequeued individual is loaded to specify the robot behavior, and after a given length of time, it is attached with the fitness value, and stored into the gene pool. The gene pool has a limited capacity, and therefore the evaluated individuals will be discarded if their fitness values are lower than the one of the worst individual in the gene pool. This mechanism realizes time sharing among many virtual individuals in each robot.

Migration procedure runs independently of the above-described GA process in each robot. An individual to be transmitted is selected (copied) from the gene pool based also on roulette wheel selection asynchronously. Each robot has a chance to send its selected individual every predefined time interval, the timing of which is randomly decided every event. The robot sends a selected individual with following probability which depends on its fitness value.

$$If(A <= C)\{ \quad P = 50(\frac{C-A}{M-A}) + 50 \quad \} \quad (1)$$

$$else \ if(A > C)\{ \quad P = 50(\frac{C-S}{A-S}) \quad \} \quad (2)$$

(P: Probability of transmitting, A: Average fitness, M: Maximum fitness, S: Minimum fitness, C: Fitness of the selected individual)
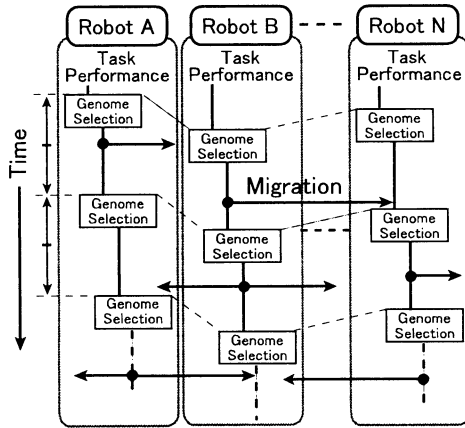
Figure 3: Asynchronous migration of individuals.

3 Preliminary Experiments

We have implemented a minimum experimental robotic system for the purpose of evaluating the proposed scheme described in the previous section. We used six Khepera miniature mobile robots (Figure 4). The issue of power supply also becomes important when realizing the Situated and Embodied Evolution paradigm. Our solution in the preliminary experiments is to adopt a power supply mechanism by which each robot moves around in a floor-and-ceiling structure and receives power continuously from a pantograph located on top of it (Figure 5). Also, a charging battery built in each robot backs up the mechanism. Infra-red communication is used for transmission of individuals between robots.
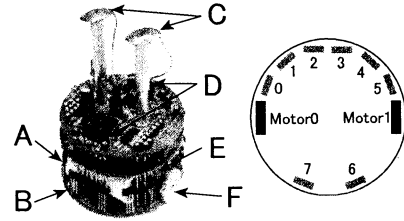
Figure 4: Khepera robot (Left - Khepera+IR communication turret A: Battery, B: Infra-red sensor, C: Pantograph, D: Infra-red emitter/receiver, E: Incremental DC motor, F: Wheel, Right - Layout of 8 infra-red sensors).
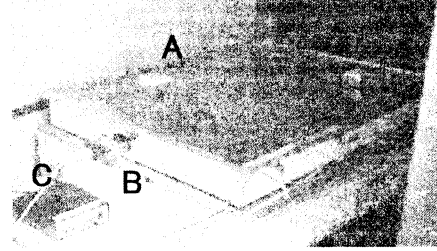
Figure 5: Experimental setup (A: Continuous power supply, B: Infra-red emitter/receiver unit, C: Power supply).

We adopted a simple two-layer neural network to control the behavior of each robot (Figure 6). The structure of the neural network (connection weights and the thresholds) was evolved by distributed genetic algorithm described in the previous section. In the neural network, 7 input nodes corresponded to 6 sensor

inputs and a threshold, each of which was expressed by 5 bit genome information. There were 2 output nodes corresponding to right and left motor outputs. So, the length of the genome was 70 bits.
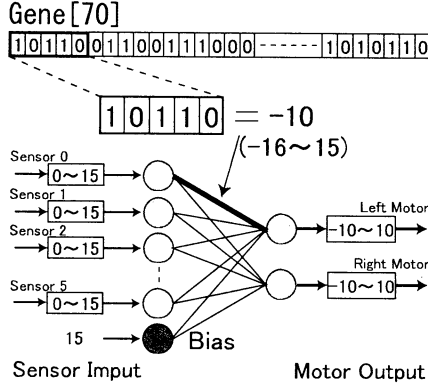
Figure 6: Relation between individual information and the structure of neural network.

Robot control programs (neural networks) for an avoidance task were evolved in the preliminary experiments for the purpose of confirming the effectiveness of the model. Each individual is evaluated by the length of movement without hitting the walls or other robots as follows:

$$fitness \mathrel{+}= SensorOff()(R\_Motor + L\_Motor), \quad (3)$$

where $SensorOff()$ returns 0 when two or more infrared sensors are activated, and otherwise returns 1, and R_Motor and L_Motor correspond to the rotation speeds of the right and left motors. Fitness value of each robot is increased at predefined time intervals when there is no input from eight infra-red sensors (which means that there are no obstacles in the neighborhood of the robot) and at least one motor is activated.

The result is shown in Figure 7, where the horizontal axis represents the total number of evaluated individuals, and the vertical axis represents the fitness. This graph shows the typical results of following 4 cases: the cases when the size of the gene pool is 1 ("1 genome per robot method"), 5 and 10, and the case in which there is no migration among robots and the size of the gene pool is 5. Average fitness of every 20 individuals is plotted in each case. We can see from this figure that the case with gene pool size of 5 with migration shows the best performance though the optimal size depends at least on the number of robots and the given task. It is also shown that migration
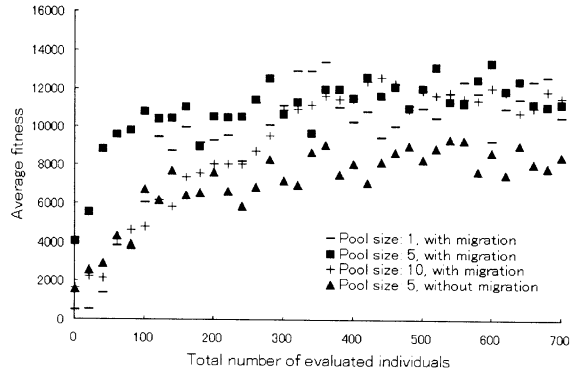
Figure 7: Evolution of robot behavior (Number of robots: 3, Mutation rate: 2%).

has a large role in this scheme. In general, there can be an unexpected difference in behavior among robots in real world. Migration of good individuals can improve the performance of the robots with unevolved gene pool.

4 Conclusion

We have proposed a distributed and asynchronous genetic algorithm for flexible and efficient robotic systems with adequate scalability that realize Situated and Embodied Evolution. We have also reported on the performance of Situated and Embodied Evolution based on the results of the preliminary experiments on the robotic system we have implemented. Further experiments will include investigation of performance evaluation of the scheme targeting at more practical tasks.

References

[1] Stefano Nolfi and Dario Floreano, *Evolutionary Robotics*, MIT Press, 2000.

[2] Richard A. Watson, Sevan G. Ficici, and Jordan B. Pollack, "Embodied Evolution: Embodying an Evolutionary Algorithm in a Population of Robots," *1999 Congress on Evolutionary Computation*, IEEE Press, 335-342, 1999.

Improvement of search ability of S-System using the limitation of age and the simplification of chromosome

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

Abstract Although the length of the function found by S -System is very short compared to that of GP, the redundancy part is still included in the function. In addition, the search ability is not enough for the various applications. In order to improve them, two strategies are proposed in this study. One is the introduction of the concept of the age, and the other is the simplification of the structure of the chromosome. By the concept of age, the search ability increases. This is because that the bugs trapped into local optimum die and the new bugs are born. For the concept of the simplification, the function -length becomes short in spite of no decrement of the fitness.

Key words: artificial life, genetic programming, evolution, chromos ome, simplification

1. Introduction

J. Koza[1] first proposed the function discovery system based on Genetic Programming (GP). However, the system has some disadvantages; (i) the schema tends to get destroyed, (ii) the solution does not stabilize and (iii) the function length becomes extremely long or extremely sort. We thus have proposed an improved system called S -System overcoming these disadvantages [2], and a method to improve the search ability is also proposed [3]. We have already applied S -System effectively in a number of search problems including the fields like design of optical elements, electronic circuit design and image recognition. In addition, in the previous paper, we proposed a non -linear optimization method incorporated S -System that significantly reduces the time for the discovery [4][5]. However, the search ability is not enough for the various applications, so we have to improve the search ability of S -System more. Although the length of the function found by S -System is very short compared to that of GP, the redundancy part is still included in the discovered function.

In order to improve them, we introduce two strategies. One is the introduction of the concept of the age, and the other is the simplification of the structure of the chromoso me. As the results, it is found that the search ability increases and the function -length is shorten.

2. Algorithm of function -discovery

The outline of the function -discovery system called S -System is mentioned here.

2.1 Main routine

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

(1) Numerous bugs with the arbitrary function are generated at random. The number **Pop** is selected from the numerous bug s in order of high fitness.

(2) The generation **Gene** of the bug is set to 0.

(3) The value of the internal energy, $energy_p$, of all the bugs is initialized to 0.

(4) The procedures from (5) to (7) are repeated for all the bugs; the bug number ranges from 1 to **Pop**.

(5) The bug p moves. This means that the values of constants K in the chromosome change slightly. That is to say, the values of K are replaced by $K + dK$, where dK is the small change of K, $\overline{K} = (K_1, K_2, \cdots, K_n)$, $d\overline{K} = (dK_1, dK_2, \cdots, dK_n)$ and n is the number of constants in the chromosome. This concept is based on Ref. (4). The details are given in Ref. (2).

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

```
                Start
(1)   Generation of initial bugs
(2)        Gene = 0
(3)   Energy_p = 0 (p=1,...,Pop)
(4)        Bug p = 1
(5)   Movement of bug p
(6)   Capture of data of bug p
(7)   Find the function ?        p = p+1
        Y                         N
       End
(4)   p=Pop ?                     N
        Y                    Gene=Gene+1
(8)   Gene=Gene_MAX ?    N    Descendant
        Y                     generation routine
       End                         (9)
```

Fig. 2 The flowchart of descendant-generation routine.

```
                SUB
(a)   Generation gap Bug No. Par
(b)        p = 0
(c)   Selection of bug by
      tournament strategy
(d)   Does the bug              N
      sexually reproduce ?
        Y                        p=p+2
(e)   Sexual reproduction   (f)
                          Asexual reproduction
(g)   Mutation
(b)   p=(Pop-Par) ?         N
        Y
      RETURN
```

(6) The bug p catches the observation data (i.e., fitness fit_p of bug p is calculated from the observation data).

(7) In the case that fitness fit_p reaches the threshold fitness Fit_{TH}, this algorithm ends. This means a bug has discovered the function f.

(8) The algorithm ends when the current generation $Gene$ reaches the maximum Generation $Gene_{max}$.

(9) After the descendant-generation-routine is called, $Gene$ is added to 1 and the algorithm returns to procedure (3).

2.2 Descendant-generation-routine

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

(a) Based on the generation-gap, Par bugs are selected and they are passed down to the next generation. The elite strategy is adopted for the generation-gap.
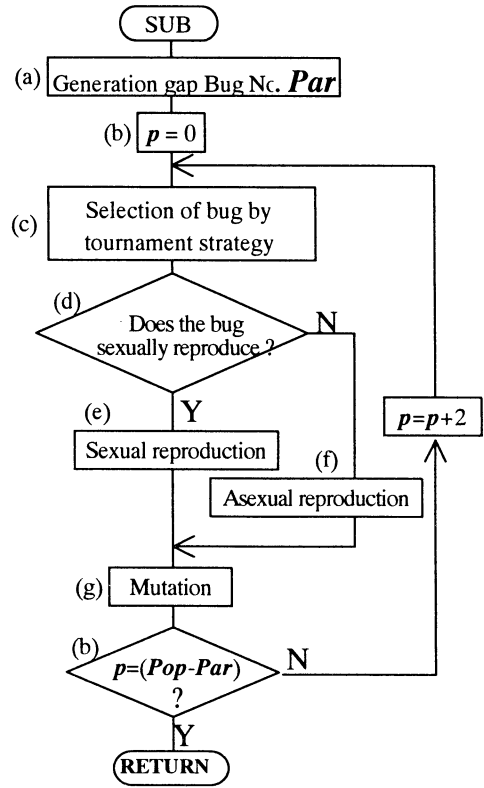
(b) The bug number p is set to be 0. By the repetition of the following procedures from (c) to (g), $Pop - Par$ bugs are generated.

(c) A bug is selected by the tournament strategy.

(d) The selected bug is judged whether it has the ability to sexually reproduce. In the case that the selected bug has the ability of sexual reproduction, procedure (e) is performed. In the other case, procedure (f) is carried out.

(e) The bug finds its partner, and they produce two children by crossover. Jump to procedure (g).

(f) Two children are produced by asexual reproduction.

(g) A part of the chromosome is changed by mutation at the rate of R_{mut}.

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

Fig. 3 the birth of new bug by the concept of the limitation of age.
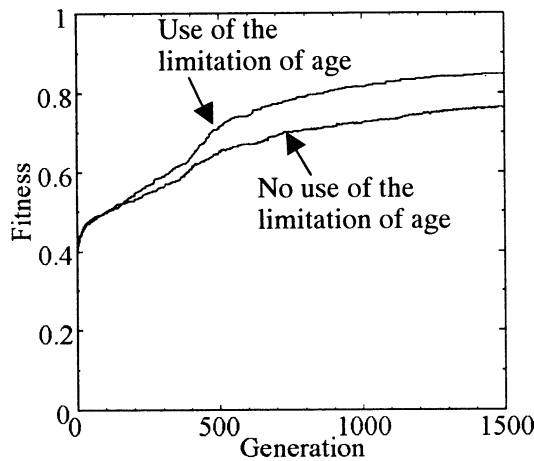
Fig. 4 The average of 30 experiments showing the relation between the generation and the fitness of the bug with the highest fitness.

3. Limitation of age

In S-System, there exist the homogeneous species and the heterogeneous species. The crossover is practiced between homogeneous bugs. As the generation proceeds, a certain homogeneity sometimes occupy most of the population, and the evolution is trapped into the local optimum. In that case, it is hard to appear the different species. For the improvement, the number of homogeneous species is limited in the S -System. Nevertheless, it is occasionally trapped into the local optimum. This is because that the bugs, which have similar chromosomes, appear and they occupy most of the population. Under such a condition, the limitation of the number of homogeneity is no longer effective. Avoiding the local optimum trap, we introduce the concept of the age.

Table 1 Comparison between GP and S -System about fitness, function length and MDL (The average of 30 experiments).

	fitness	function length	MDL
GP	0.8139	1706.0	1433.4
S-System	0.8238	43.1	35.5

When a bug is born, the age of the bug is set to be 0, and if the mutation is not practiced at crossover, the age of the bug increases with generation. When the age of the bug exceeds a certain age (threshold age), the bug dies. The outline is shown in Fig. 3. In this figure, the threshold age is set to be 100. As understood from the left side of Fig. 3, the bug dies when the age exceeds 100, and a new bug is born at the next generation.

The age is reset to be 0 at the mutation. In S-System, the bug with different chromosome is regarded as the heterogeneous species. When a heterogeneity is produced by mutation, the age is set to be 0. The outline is shown in right side of Fig. 3.

Using this strategy, we tied to practice 30 experiments. The result is shown in Fig. 4. In the experiments, the observation data in Ref. (4) is used. The threshold age is set to be 200. From this figure, it is understood that the fitness of the model incorporating the limitation of age is higher than that of no limitation of age. This is because that the bugs in local optimum die because of the limitation of age. As the result, the probability of the trap into the local optimum is reduced.

4. Simplification of chromosome

The other improvement is the reduction of the redundancy of function length. Here, we introduce the concept of the simplification of chromosome. Table 1 shows the comparison between GP and S-System about fitness, function length and *MDL*. From this table, it is understood that the function length by S-System is very short compared to that of GP. However, the redundancy part is still included in S -System. As shown in Fig. 5, the function generated by S -System is composed of some nodes. When the observation data (x, y, ...) are substituted for a node, there sometimes happens that the value of the node does not change
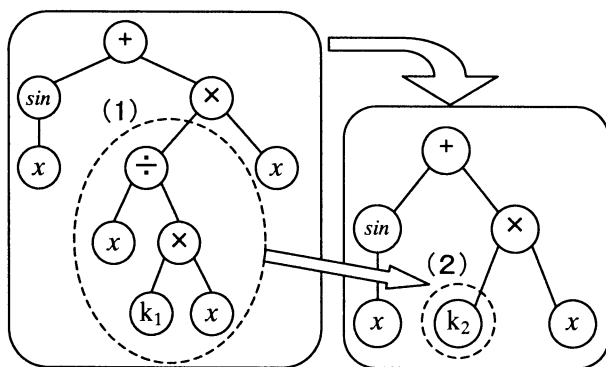
(a) before simplification   (b) after simplificatio n

Fig. 5 An example of the simplification of chromosome.

before simplification   after simplification
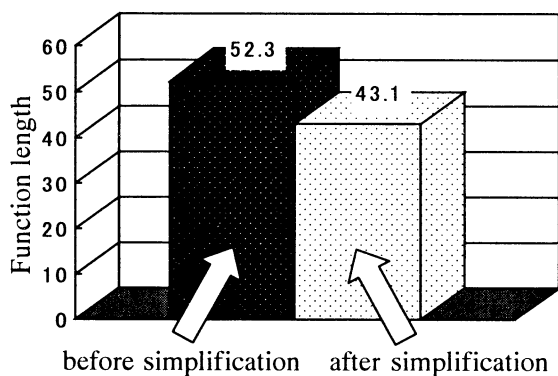
Fig. 6 The average of function -length at 30 experiments.

at all. In Fig.5 (a), the value of the part (1) does not change at any value of x. In this case, the node and the lower nodes are replaced with a constant value. This is shown as the part (2) in Fig. 5 (b). By this strategy, the function lengt h is shorten but the meaning of the function does not change.

The result is shown in Fig. 6. The function -length using the simplification is shorter than that of no -simplification. As far as the fitness concerned, the average-fitness of 30 experiments is 0.78 and 0.77 for the simplification and no-simplification, respectively. To sum up, for the concept of the simplification, the function -length becomes short in spite of no decrement of the fitness.

5. Conclusions

To improve the search ability of S -System, two strategies are proposed in this study. One is the introduction of the concept of the age, and the other is the simplification of the chromosome. By the concept of age, the search ability increases. This is because that the bugs trapped into local optimum die and the new bugs are born. For the concept of the simplification, the function -length becomes short in spite of no decrement of the fitness.

References

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

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

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

(4) T. Shimmomura, K.Yamashita and S.Serikawa "Function discovery system model using non-linear optimization method ", *Proc. Of 6th Symp. On Artificial Life and Robotics (AROB 6th '01)*, pp.321-324 (2001).

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

Optimization of Multi-objective Function based on The Game Theory and Co-Evolutionary Algorithm

Ji-Yoon Kim, Dong-Wook Lee, and Kwee-Bo Sim

School of Electrical and Electronic Engineering, Chung-Ang University, Korea
221, Huksuk-Dong Dongjak-Ku, Seoul 156-756, Korea
jonathan@jupite.cie.cau.ac.kr   kbsim@cau.ac.kr

Multi-objective Optimization Problems (MOPs) are met by engineers more frequently than generally thought when they try to solve engineering problems. In the real world, the majority cases of optimization problems are the composed problems of several competitive objective functions [1]. In this paper, we introduce several approaches to solve MOPs. In the introduction, established optimization algorithms based on the concept of Pareto optimal set are introduced. Contrary these algorithms, we introduce theoretical backgrounds of Nash Genetic Algorithm (Nash GA) and Evolutionary Stable Strategy (ESS), which are based on Evolutionary Game Theory (EGT). Moreover ESS is the basis of co-evolutionary algorithm proposed in this paper. Then in the next chapter, the explanation about the architecture of Nash GA and Co-evolutionary algorithm for solving MOPs follows. In the final chapter from the experimental results we confirm that two algorithms, Nash GA and co-evolutionary algorithm, based on Evolutionary Game Theory (EGT) can search optimal equilibrium strategy of MOPs.

Keywords: Multi-objective optimization problem, Pareto optimality, Nash GA, Evolutionary Stable Strategy, Co-evolutionary algorithm, Game Theory

1. Introduction

Most of the real world problems that are came in contact with engineers includes simultaneous optimization of more than two competitive objective functions. In the case of Single-objective Optimization Problems (SOPs) having a single objective function, the optimal solution is obtained clearly but this does not hold good in the case of MOPs. In this case, instead of a single optimal solution, there exists a set of solutions generally known as Pareto optimal solution. These namely 'non-dominated' solutions are optimal in the sense that no other solutions in the search space are superior to them when all objectives are considered. With these concepts, Goldberg proposed Pareto Genetic Algorithms (Pareto GAs) that have became a sort of standard method [2]. Then the approach based on the concept of Pareto optimal solution has been studied abundantly and several techniques for applying Genetic Algorithms (GAs) have been developed like as 'Pareto Ranking', 'Fitness Sharing'. But in this paper, we introduce searching techniques based on Evolutionary Game Theory (EGT) for MOPs. These techniques are different from established approaches. So we propose new applying method to solve MOPs with co-evolutionary algorithm.

The first trial has been founded by J. F. Nash to solve MOPs applying EGT in the early 1950's. This theorem introduced the concept of 'Game Player' to search the equilibrium strategies of MOPs came from Game Theory and economics. As noted by Nash, during the game each player pursues optimal strategy for his objective function and finds the optimal equilibrium strategies of MOPs through the competition between themselves. This solution is so called 'Nash Equilibrium'. To apply these ideas to solve MOPs using GAs, M. Sefrioui suggested 'Nash GA' in 1998. Nash GA is composed of the combination of Nash's theorem and Genetic Algorithms (GAs). This algorithm searches 'Nash Equilibrium Point' as the solution of MOPs through an evolutionary process.

Ever since the existence of Nash equilibrium points has been introduced, Game theorists started on study about searching algorithms for evolutionary equilibrium point. The main idea of Evolutionary Game Theory (EGT) is that not 'a rational strategy selection of each game participants' but through 'the Darwinian selection process', each population of a game can find Evolutionary Stable Strategy (ESS) as an optimal equilibrium solution of game. Thanks to these ideas that MOPs modeled through the evolutionary game have at least one or more ESS, we propose optimal equilibrium strategy searching method using co-evolutionary algorithm. In the interests of this, we suggest co-evolution architecture including game model with the adaptation of co-evolutionary algorithm previously established and perform experimental evaluations.

2. Nash Genetic Algorithms

Since mathematical basis founded by Von Neumann in the late 1920s', Game Theory has contributed to the study for solving MOPs that are indulged in the sphere of mathematics and economics. Game Theory introduces the notion of games and players associated to an optimization problem. In the case of a multi-objective design through Game Theory, each player involved is named player and has his own criterion. During the game, namely Nash Games, they try to improve it until the system reaches equilibrium.

2.1 Nash Equilibrium

Nash equilibrium is the solution of a non-cooperative strategy for MOPs first introduced by J. F. Nash in 1952. By Nash, each participant of game has his own strategy set and objective function. During the game each player searches

optimal strategy for his objective function while strategies of others are fixed. Nash frequency σ presents how frequently game strategy is changed by participants. Generally $\sigma = 1$, this means that the exchange of best strategies takes place at the end of each generation. In this frame evolutionary game is conducted and when no player can further improve his criterion, the system has reached a state of equilibrium named Nash equilibrium [3].

2.2 Nash Genetic Algorithm

The idea is to bring together genetic algorithms and Nash strategy in order to make the genetic algorithm build the Nash Equilibrium (Sefrioui, 1998). In the followings, we present how such merging can be achieved with 2 players trying to optimize 2 different objectives.

Let $s = XY$ be the string representing the potential solution for a dual objective optimization problem. Then X denotes the subset of variables handled by Player 1 and optimized along criterion 1. Similarly Y denotes the subset of variables handled by Player 2 and optimized along criterion 2. Thus, as advocated by Nash theory, Player 1 optimizes s with respect to the first criterion by modifying X while Y is fixed by Player 2. Symmetrically, Player 2 optimizes s with respect to the second criterion by modifying Y, while X is fixed by player 1.

The next step consists in creating two different populations, one for each player. Player 1's optimization task is performed by population1 whereas Player 2's optimization task is performed by population2.

Let X_{k-1} be the best value found by Player 1 at generation $k-1$ and Y_{k-1} the best value found by Player 2 at generation $k-1$. At generation k, Player 1 optimizes X_k while using Y_{k-1} in order to evaluate s (in this case, $s = X_k Y_{k-1}$). At the same time, Player 2 optimizes Y_k while using X_{k-1} in order

to evaluate s (in this case, $s = X_{k-1} Y_k$). After the optimization process, Player 1 sends the best value X_k to Player 2 who will use it at generation $k+1$. Similarly, Player 2 sends the best value Y_k to Player 1 who will use it at generation $k+1$. Nash equilibrium is reached when neither Player 1 nor Player 2 can further improve their criteria [4].

3. Co-evolutionary algorithm

3.1 Evolutionary Stable Strategy

Evolutionary Stable Strategy (ESS) proposed by a biologist having a scholar of world-wide fame Maynard Smith based on Evolutionary Game Theory (EGT) and defined as an unchangeable strategy by other strategies if almost of the population take this. This means that no matter how excellent strategy may be can't make a profit perfectly and stay predominant over other inferior strategies permanently. After all it means that in real ecosystem more evolutionary stable species can be reserved than superior species and in other worlds an evolution chooses the strategy that not only executes progressive direction but also make move equilibrium state.

In the next paragraph from the idea that ESS can be found through the game how the equilibrium strategy can be searched is explained.

3.2 Architecture of co-evolutionary algorithm

To design co-evolutionary algorithm based on Game Theory the first we establish game player with randomly generated populations. All individuals in the each population are provided 'funds' that will be used through the game. During the game each individuals in the first population plays the game with others in the rest populations and is paid the gain calculated by Equation (3) and (4). Other individuals in the rest populations execute the game in the same manner by turns. Taking adapted funds as fitness, next generation individuals are produced in each population independently through crossover and mutation.
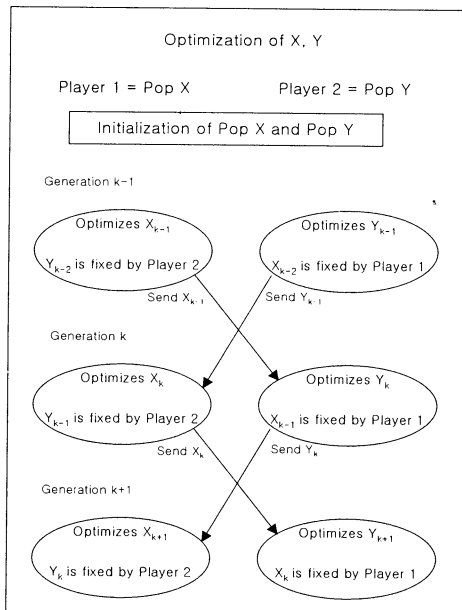
Fig. 1. Strategy of Nash Genetic Algorithm
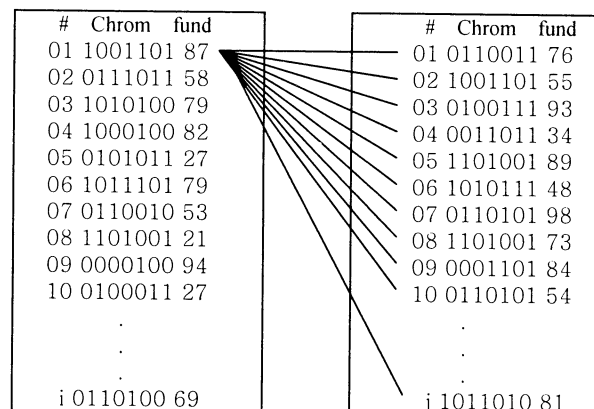
Fig. 2. Individuals of each population

Where # denotes the number of individual and Chrom means bit-string of each individual.

The example of MOPs having two variables is given to help

(1) Two populations are randomly generated like Fig. 2.

(2) Each individual is provided one-priced initial funds.

(3) The initial individual of the first population plays with each individual in the other population and is rewarded gain produced through the game by turns.

(4) The process of (3) is executed for all individuals of the first population one by one.

(5) The processes of (3) and (4) are executed for all individuals of the second population analogously.

$$fund_{t+1}(i) = fund_t(i) + \frac{\sum_{n=1}^{j} gain_t(i,n)}{j} \quad (1)$$

$$gain(i,j) = f_a(x_j) - f_a(x_i) + f_b(x_j) - f_b(x_i) \quad (2)$$

In Equation (1) i means the number of individual of the first population and j means the number of the second population. In Equation (2) we label $f_a(x_j)$, $f_a(x_i)$, $f_b(x_j)$, and $f_b(x_i)$ with the values calculated when $x = x_i$ or x_j for $f_a(x)$, $f_b(x)$ from Fig. 3.
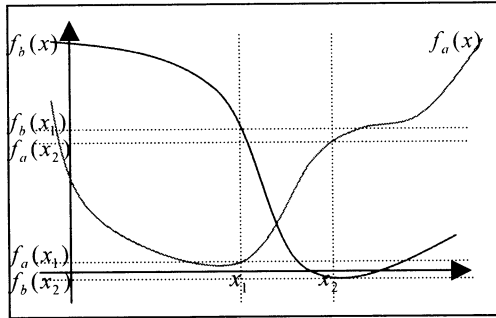
Fig. 3. Graph of objective function of gain calculation

(6) Using $fund_{t+1}(i)$ and $fund_{t+1}(j)$ determined from the previous procedures, each population produce next generation individuals independently.

(7) Until ending condition is satisfied the procedures from (3) to (6) are reiterative.

4. Experiments and Results

4.1 Experiments on Sefrioui's function

In the first experiment, we solve applying Nash GA to MOPs proposed in Sefrioui's paper.

$$f_1(x,y) = (x-1)^2 + (x-y)^2$$
$$f_2(x,y) = (y-3)^2 + (x-y)^2 \quad (3)$$
where $-5 \le x, y \le 5$

To solve MOPs showed above using genetic algorithms, we allotted two populations to each objective function for searching solutions and applied elitism. Table 1 shows the most optimized solutions for this problem by using Nash GA and co-evolutionary algorithm. Hence X_1, Y_1, F_1 are the optimized values by population 1 and X_2, Y_2, F_2 are the optimized values by population 2. The optimized values by Sefrioui using Nash GAs are the same values found by Sefrioui in his paper. This presents Nash equilibrium point. Fig. 4 shows the change of f_1, f_2 values for searching process by using Nash GAs and Fig. 5 shows the change of f_1, f_2 values for searching process by using co-evolutionary algorithms. From the results, we can find that co-evolutionary algorithms search more efficient solutions than the solution found by Nash GA and these solutions are exist in the boundary of Pareto frontier too.

Table 1. The most optimized values for Sefrioui's function, which are found by using each algorithm

	X_1	Y_1	F_1
Nash GA	1.6673	2.3297	0.8840
	X_2	Y_2	F_2
	1.6697	2.3393	0.8849
Co-evolutionary algorithm	X_1	Y_1	F_1
	1.8187	2.1810	0.8016
	X_2	Y_2	F_2
	1.7564	2.4696	0.7810
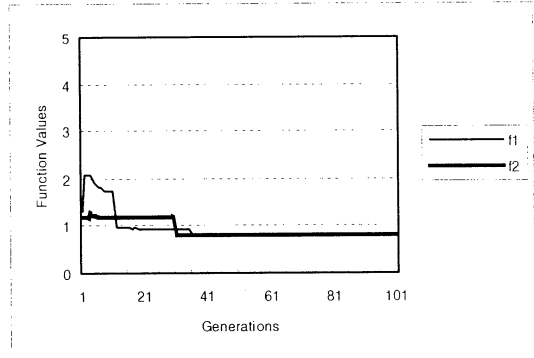
Fig. 4. Change of f_1, f_2 by Nash GA
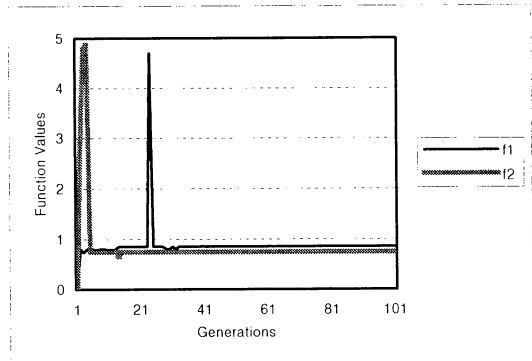
Fig. 5. Change of f_1, f_2 by Co-Evolutionary algorithm

4.2 Experiments on Schaffer's function

The second experiment is to solve MOPs proposed by Schaffer [5].

$$f_1(x,y) = (x-1)^2 + (y-1)^2$$
$$f_2(x,y) = (x-4)^2 + (y-4)^2 \qquad (4)$$

where $-5 \leq x,y \leq 5$

Like as the previous experiment, we allotted two populations for each objective function. The experimental results searched by applying each algorithm to this problem are shown in table 2, Fig 6 and Fig 7. As see in these results, this point exists in Pareto optimal front like previous experiment moreover co-evolutionary algorithms have found more optimized solution for this problem.

Table 2. The most optimized values for Schaffer's function, which are found by using each algorithm

	X_1	Y_1	F_1
Nash GA	0.9933	4.0004	9.0023
	X_2	Y_2	F_2
	0.9933	4.0004	9.0403
Co-evolutionary algorithm	X_1	Y_1	F_1
	2.3499	2.6825	4.6531
	X_2	Y_2	F_2
	2.4994	2.4641	4.6107
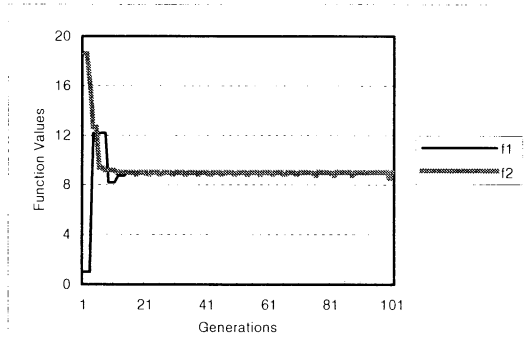
Fig. 6. Change of f_1, f_2 by Nash GA
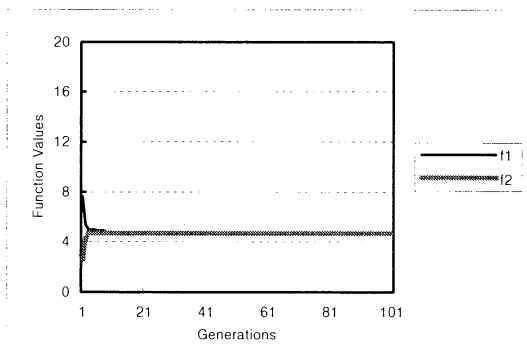
Fig. 7. Change of f_1, f_2 by Co-Evolutionary algorithm

From the previously executed experimental results we confirmed that it is regarded as appropriate to search optimal equilibrium strategy by applying co-evolutionary algorithm to Evolutionary Game Theory model and this could be regarded as the solutions of MOPs. Especially we found that co-evolutionary algorithm based on EGT model could search more optimized solution.

5. Conclusions

In this paper we introduced Nash GA and co-evolutionary algorithm that approach different way from previously established. These algorithms search the optimal equilibrium strategy of MOPs through the Evolutionary Game Theory. And applying these algorithms to solve MOPs, we debated about the propriety and efficiency of these algorithms. Another fact found through the experiments is that in the case of Nash GA to keep from the premature convergence to local minimum, Distance Dependent Mutation method, which uses alternative mutation rate according to Hamming distance of parents in generating children individuals from the parents, must be used [6]. Differently from this, in the case of co-evolutionary algorithm we confirmed that more optimized equilibrium strategy could be found nevertheless only simple mutation method, generally used, is applied.

Acknowledgements

References

[1] Eclart Zitzler, *Evolutionary Algorithms for MultiObjective Optimization: Methods and Applications,* A dissertation submitted to the Swiss Federal Institute of Technology Zurich for the degree of Doctor of Technical Sciences, November 11, 1999.

[2] M. Sefrioui and J. Periaux, "Nash Genetic Algorithms : examples and applications," *Proc. of the 2000 Congress on Evolutionary Computation CEC00,* IEEE Press, pp. 509-516, 2000.

[3] Nash J. F., "Noncooperative games," *Annals of Mathematics,* 54:289, 1951.

[4] J. Horn, N. Nafpliotis, and D. E. Goldberg, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization," *Proc. of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence(ICEC '94),* vol. 1., NJ, IEEE Service Center, pp. 82-87, 1994.

[5] David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning,* New York, Addison-Wesley, pp. 199-201, 1989.

[6] A. Borodin., B. Mantel., S. Peigin., J. Periaux., and S. Timchenko., "Application of the Genetic Algorithm for Heat Flux Optimization Problem," *EUROGEN97 (Trieste University, Italy* 1997.

A Study on Compensation of Modeling Errors of Evolutionary Robot

KITABATAKE Shin
Information Engineering,
Mie University
1515 Kamihama-cho, Tsu, Japan

FURUHASHI Takeshi
Information Engineering,
Mie University
1515 Kamihama-cho, Tsu, Japan

Abstract

This paper presents a framework for simultaneous evolution of robot structure and behavior. This method utilizes computer simulation for drastically reducing the number of trials in the actual environment. Modeling errors are estimated in the simulated environment using evolution strategy. Under the modified simulation environment, optimal structure and behavior are sought. Experiments with LEGO MINDSTORMS are done to demonstrate the effectiveness of the proposed method.

1 Introduction

Evolutionary robotics that evolves behavior and structure of robots has been studied [1][2]. It is expected that the robots can acquire their proper structure and behavior for given tasks under given environment by the evolution. Since a great amount of trial and error is needed for the evolution, it is so time consuming if it is done in the actual environment. Robots would be worn out. Reduction of trial in the actual environment is one of the major topics of evolutionary robotics. Floreano and Urzelai [1] introduced an incremental evolution for tackling complex problems that cannot be evolved from scratch. This could reduce the number of trials. Computer simulation is another approach to the above difficulty. Robots and the environment are modeled in a virtual world. Because of remarkable advance in computer technology, simulation has become one of the major tools for scientific research and engineering. A new difficulty here is modeling error. The evolved robots in the virtual world have often been unable to act properly in the actual environment. Lund et al. [2] have tried to modify the error by incorporating deviations of individual sensors. Kondo et al. [3] proposed a robust controller by evolving the learning algorithm of the neural network controller. This controller is quick in changing the synapses under unknown environment. However, these studies have not considered evolution of robots' structure and behavior simultaneously.

This paper presents a framework for simultaneous evolution of robot structure and behavior by estimating and compensating the modeling errors of evolutionary robots. The proposed method estimates the errors utilizing the information about the deviation of behavior between the simulation and the actual experiment using evolution strategy. Under the modified simulation environment, robots' evolution is re-carried out. This method drastically decreases the amount of trial and error in the actual experimental set-up. Appropriate structure and behavior can be evolved even under the changing environment. Experiments with LEGO MINDSTORMS are done to demonstrate the effectiveness of the proposed method.

2 Proposed Method

Simulated environment should be created as close as possible to the actual environment. In the case where the real world is well set up, modeling errors would be negligible. But in many cases, the actual world is so complex, and the modeling errors make the deviation between the behavior in the simulation and the experiment very large.

The performance P and the parameters X of the robot can be expressed as

$$P = f(X) \qquad (1)$$

where f is the model of the environment and the robot. If the performance in the actual environment is different from that in the simulation, then the problem here is to find the deviated parameters from the difference in the performance. This is an inverse problem. Since the function f is non-linear, effective methods for solving this problem would be search algorithms, such as evolution strategy.

The procedure for this parameter search in this paper is given as follows:

(1) An autonomous robot is mounted with a simulator with a simulated environment. The robot searches for the best structure/behavior for fulfilling a given task in the simulated environment.

(2) The robot with the best parameters acts in the real environment. The actual behavior would be different from the best one in the simulation because of modeling errors.

(3) The modeling errors are estimated in the simulated environment by searching for the structure/behavior that give the same performance obtained at step (2).

(4) The simulated environment is modified in accordance with the estimation, and the best structure/behavior are re-searched with the modified model.

The modeled environment is made to be a virtual one, and the best solution in virtual system is also valid in the actual environment.

3 Experiments

3.1 Experimental set-up

In this paper, we carried out experiments using a simple line trace problem as shown in Fig. 1. The robot was a "LEGO MINDSTORMS". The floor was a sheet of paper. The line was a rectangle with round corners. The length was 67.5 [cm], and the width was 44.5 [cm]. The configuration of the robot is shown in Fig. 2. The length and the width, and the height of the robot were 14, 18, and 12 [cm], respectively. It had two wheels and two light sensors. The light sensors detect black line drawn on the floor. The robot turns right if the sensor on the right hand side detects the black line, and left if the left one does. The one step takes 30ms. The simulated environment was on a PC.

The parameters and their variable ranges of the robot were:

R_{LD}: distance between the light sensors   2~20 [cm]
R_{LC}: distance between the center of the robot front and the middle of the light sensors   0~10 [cm]
R_L: length of the robot   10~20 [cm]
R_D: degree of rotation   8.2~81.8[deg]

The middle point of the light sensors deviates to the right hand side by R_{LC}. The robot rotates clockwise by R_D degrees at one step. At each step, the robot takes one of the actions: move forward, rotates clockwise, or rotates counterclockwise.

The optimal parameters were those to give quickest run around the circle in Fig.1 twice.

The performance index used for the evolution strategy was defined as

$$G = w_1|T_s - T_r| + w_2|N_s - N_r| + w_3|H_s - H_r| \qquad (2)$$

where T_s [sec] and T_r [sec] were the running times in the simulated and the real environment, respectively, N_s and N_r were the numbers of rotation in the simulated and the real environment, respectively, H_s and H_r were the histories of rotation in the simulated and the real environment, respectively, and w_1, w_2, w_3 were weights. The history of rotation was recoded as a sequence of binary bits where 1 means a turn to the right and 0 to the left. This bit string was treated as a binary number. The turns in the early stage of running were more influential than those in the later stage, thus the bits in the early stage was put on the higher figures of the binary number. The weights w_1, w_2, and w_3 were set at 10, 10000, and 1, respectively in this experimental set-up.

The following variables are assumed to be the candidates for the modeling errors:

E_1: advancing speed [cm/sec]   (-5~5)
E_2: rotation speed [deg/sec]   (-40~40)
E_3: rotation speed while the robot is moving forward [deg/sec]   (-3~3)
E_4: angle at the initial position [deg] (-3~3)
E_5: distance between the light sensors [cm](-0.5~0.5)
E_6: length [cm]   (-0.5~0.5)
E_7: dead time by friction before rotation [sec]   (0~0.1)
E_8: inertial movement forward before rotation [sec](0~0.2)

The ranges in the parentheses were considered to be the maximum variable ranges.

Since it was difficult to make all the conditions the same in the actual environment, we recorded one trial and regarded it as the result in the real environment.

3.2 Compensation of modeling errors

3.2.1 Optimal parameters in simulated environment

At the first step of the experiment, we searched the best structure/behavior using the simulated environment as shown in table 1. We used the evolution strategy for the search. The population size was set at 300, and the maximum generation was set at 200. The track of the robot with the optimal parameters obtained in the simulation is shown in Fig.3. We did an experiment using the parameters that were optimal in the simulation. The performances are listed in Table 2. The running times and the numbers of rotation in the simulated/actual environment were different. The track of the actual robot is shown in Fig. 4. The tracks in Figs 3 and 4 were quite different.

44.5 [cm]

67.5 [cm]
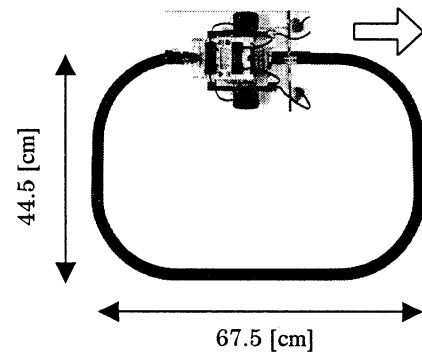
Figure 1: Line trace problem

Light sensor

R_L [cm]

Light sensor

R_H [cm]
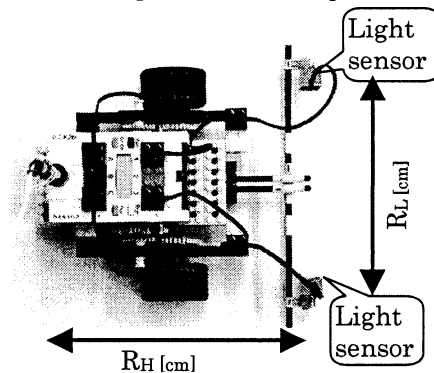
Figure 2: Initial shape of the robot

Table 1 : Best structural/behavior parameters in the simulated environment

Distance of light sensors (R_{LD})[cm]	11.7
Deviation of light sensors (R_{LC}) [cm]	0.0
Length of robot (R_L) [cm]	19.9
Degree of rotation (R_D) [deg]	41.2

Table 2: Performances in the simulated/actual environment with the parameters in Table 1

	Running time [sec]	Number of rotation
Simulator	48.6	16
Real world	46.9	23
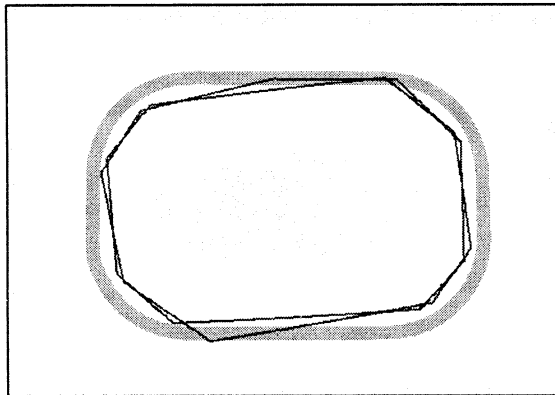
Figure 3: Track with best parameters in simulated environment
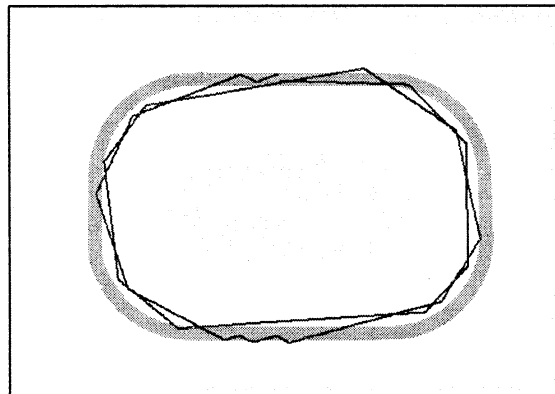
Figure 4: Track using the parameters in Table 1

3.2.2 Estimation of modeling errors

Modeling errors were estimated from the difference in behavior in the simulation and the actual experiment in 3.2.1. The estimation was carried out in the simulated environment by searching for combinations of the candidate parameters E_1-E_8 that gave the same behavior as in the experiment. We used the evolution strategy as the search engine. The population size was set at 500, and the stopping condition was at the 200-th generation. Four sets of parameters were obtained by repeating the search with the evolution strategy four times. The random seed

was changed for each evolutionary search. The results are shown in Table 3.

3.2.3 Re-search for optimal parameters in the modified environment

The simulator was modified by the estimated errors in Table 3, and the optimal structural/behavior parameters were re-searched. Each set of obtained parameters was supposed to be optimal also in the actual environment. They were implemented on the actual robot and tested. The results are shown in Table 4. The running times T_s and T_r in the simulated/real environment, and the numbers of rotation N_s and N_r in the simulated/real environment coincided with each other in the case with the 3-rd set of parameters. The track in this case is shown in Fig. 5. The optimal behavior was achieved also in the actual environment.

Table 3: Estimated modeling errors

	1st	2nd	3rd	4th
E_1 [cm/s]	1.87	1.94	2.06	2.29
E_2 [deg/sec]	-0.02	-4.85	1.51	-0.98
E_3 [deg/sec]	0.13	0.99	-0.61	0.04
E_4 [deg]	1.83	0.85	2.06	2.80
E_5 [cm]	-0.24	-0.21	-0.19	-0.37
E_6 [cm]	0.24	0.11	-0.34	0.07
E_7 [sec]	0.03	0.04	0.03	0.04
E_8 [sec]	0.06	0.07	0.06	0.07
T_s [sec]	46.9	47.0	47.0	47.0
N_s	23	23	23	23
G	570	90	90	750

Table 4: Parameters supposed to be optimal and the performance in the real environment

	1st	2nd	3rd	4th
R_{LD} [cm]	14.93	14.23	14.98	14.82
R_{LC} [cm]	0.0	0.03	0.0	0.0
R_L [cm]	19.93	19.79	19.29	19.88
R_D [deg]	57.44	55.43	57.95	57.73
T_s [sec]	42.3	42.3	42.2	41.6
N_s	12	12	12	12
T_r [sec]	48.0	53.5	42.0	44.7
N_r	20	28	12	16
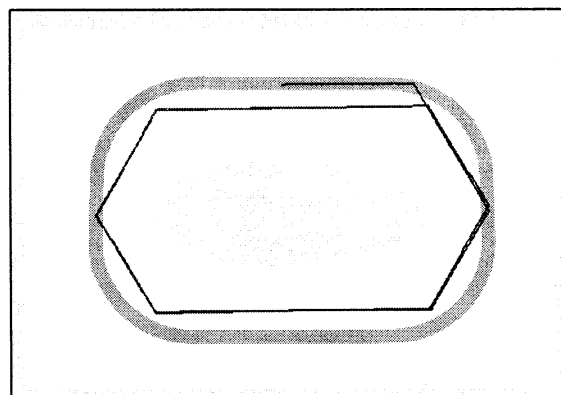
Figure 5: Optimal track of robot with the compensated errors

3.2.4 Another environment

We applied this method to the environment in Fig.6. The line was drawn on the floor of vinyl tile instead of paper. The optimal structure/behavior obtained in the simulation is listed in Table 5. The population size was 300, and the stopping condition was at the 200-th generation. The optimality is defined as the shortest lap around the line in Fig.6. Table 6 shows the comparison of the performances in the simulation and the experiment using the same parameter in Table 5. Four candidate sets of modifications were generated by the evolution strategy considering the errors in Table 6, and the performances incorporating the supposed errors are compared in Table 7. The 1-st set of candidate parameters was able to compensate the modeling errors and the performances coincided.

It should be noted that the number of trials using the actual experimental set-up was only five in each case in 3.2.1-3.2.3 and in 3.2.4. Major part of trial and error was done in the simulator.

30 [cm]
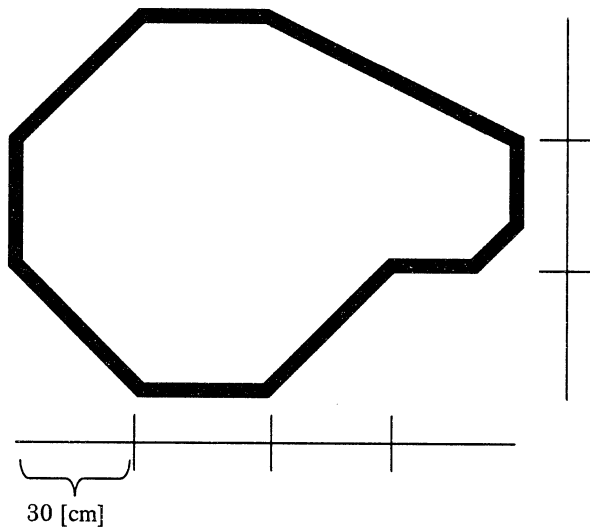
Figure 6: Another line trace problem

Table5: Best structure/behavior in the simulated environment in the case in Fig.6

Distance of light sensors (R_{LD})[cm]	17.6
Deviation of light sensors (R_{LC}) [cm]	0.0
Length of robot (R_L) [cm]	19.7
Degree of rotation (R_D) [deg]	33.7

Table 6: Performances in the simulated/actual environment with the parameters in Table 5

	Running time [sec]	Number of rotation
Simulator	34.4	14
Real world	41.9	18

Table 7: Parameters supposed to be optimal and the performance in the real environment in the case in Fig.6

	1st	2nd	3rd	4th
R_{LD} [cm]	15.57	16.91	18.29	12.03
R_{LC} [cm]	0.02	0.18	0.01	0.02
R_L [cm]	19.69	19.59	19.73	19.98
R_D [deg]	29.2	25.5	24.93	20.29
T_s [sec]	37.0	37.6	37.3	37.9
N_s	13	13	13	18
T_r [sec]	37.6	39.2	38.0	38.4
N_r	13	16	13	19

4 Conclusion

Compensation of modeling errors is one of the major difficulties in the field of evolutionary robotics. In this paper, we proposed a framework for evolution of robot structure and behavior by estimating and compensating the modeling errors of evolutionary robots, and demonstrated the effectiveness of the proposed method. This method was able to reduce the amount of trial and error in the actual environment drastically.

In the future, we are planning to increase the number of error variables, and investigate more efficient algorithm for this problem.

References

[1] Floreano, D. and Urzelai, J.:Evolutionary Robotics The Next Generation, In T.Gomi(ed), Evolutionary Robotics III, Ontario(Canada):AAI Books, 2000.

[2] H.H.Lund, et al.:Evolving Robot Morphology, Proc. of IEEE Fourth Int'l Conf. On Evolutionary Computation, IEEE Press, NJ, 1997.

[3] Kondo, et al.:Realization of Robust Controllers in Evolutionary Robotics -- Proposal of a Neural Network with Dynamically-Rearranging Function --, Trans. of SICE, Vol.35, No.11, pp.1407-1414, 1999.

The recognition of the dynamic system fuzzy model

Fan Ding

Deparment of Computer Engineering   Qingdao institute of Architectural Engineering,

Address: 2 Changjiang zhonglu road, Qingdao Development Zone of Economy and Technology

Qingdao of China

Postal Code :266520,

Email:wwwdingfan@163.com

Abstract:

The article discusses the fuzzy model application in systemic recognition, and introduces that systemic recognition is to build up a system equivalent to the one to be recognized on basis of the measuring of the input/output data of a defined type of system. The article introduces a method for fuzzy model recognition for dynamic systems.

Keyword: fuzzy model, recognition, membership function, clustering, fuzzy partition

1.Introduction

The building of a systemic precise mathematical model is to describe through certain mathematical formulation the inner link of the input and output of a system which we've found out through mathematical treatments on mass data take from the target system. However, the point is, a precise mathematical model of a complicated controlled plant or course is usually difficult to build or even impossible to do so.

With the application of fuzzy set theory, building up fuzzy models of the systems of which precise mathematical model is difficult to build, has become another virtual approach of systemic recognition.

This article intends to introduce a method of fuzzy model recognition of dynamic systems.

2. Problem description

Let's study a fuzzy system as follows:

$$y(t) = [y(t-1) \circ \cdots \circ y(t-n_y)] \circ$$
$$[u_1(t-\tau_1) \circ u_1(t-\tau_1-1) \circ \cdots \circ$$
$$u_1(t-\tau_1-n_1) \circ \cdots \circ [u_m(t-\tau_m) \circ$$
$$u_m(t-\tau_m-1) \circ \cdots \circ u_m(t-\tau_m-n_m)] \circ R$$
$$\text{———————} \quad (2\text{-}1)$$

Where y(∘) is the output fuzzy variable, $u_1(\circ)$, …, $u_m(\circ)$ is the input fuzzy variable, R is the relation based on the reference set, " ∘ " is the composition operator based on the reference set. Factorial n_y , n_1 , …, n_m and lag coefficient τ_1 , …, τ_m are systemic structural parameter. Job required includes structural recognition and the recognition of R.

In order to ease the expression, the above can also be:

$$X_1(t)=y(t-1)$$
$$\cdots$$
$$x_{n_y}(t) = y(t-n_y) ,$$
$$x_{n_y+1}(t) = u_1(t-\tau_1),$$
$$\cdots$$
$$x_n = u_m(t-\tau_m-n_m),$$

Where $n = \sum_{i=1}^{\infty}(n_i + 1)$. Thus, （2-2）can be in the form of

$$y(t) = x_1(t) \circ \cdots \circ x_n(t) \circ R \quad \text{———} \quad (2\text{-}2)$$

Where y, x_1 , …, x_n are common component of discrete universes of discourse Y, X_1, …, X_n. Supposing each universe of discourse has the same number of reference fuzzy sets, take these sets as:

$$\begin{bmatrix} A_{i1},\cdots, A_{ir_i} \in F(X_i), & i=1, n \\ B_1,\cdots, B_r \in F(Y) \end{bmatrix} \quad \text{———} \quad (2\text{-}3)$$

F(Y) stand for all the fuzzy sets of set{Y}, else the same. Reference set can be defined as semantic of the universe of discourse, to define the reference set is to define its membership function, to guarantee the performance of the fuzzy model, it is required that all the reference sets should be normal convex sets, and satisfy the following requirements:

$$\begin{bmatrix} A_{i1},\cdots,A_{ir_i};X_i \to [0,1] & i=1,n \\ B_1,\cdots, B_r; Y \to [0,1] \end{bmatrix} \quad \text{———} \quad (2\text{-}4)$$

$$\begin{bmatrix} \forall x_i \in X_i, & \exists j \in r_i, & A_{ij}(x) > 0 & i=1, n \\ \forall y \in Y, & \exists l \in r, & B_l(y) > 0 \end{bmatrix} \quad \text{———} \quad (2\text{-}5)$$

Thus, it is possible to figure out a suite of feasiblity distribution of each fuzzy set of the universe of discourse based on the reference fuzzy set.

$$
\begin{cases}
P_{ij} = \sup \min[A_{ij}(x_i),\ X_i\,(x_i)] & i=1;\ j=1,r_i \\
P_l = \sup \min[B_1,\ Y\,(y)\,] & l=1,r
\end{cases} \quad\text{(2-6)}
$$

Specially, when Xi and Y are single-point fuzzy set, formula can be simplified as:

$$
\begin{cases}
P_{ij} = A_{ij}(x_i) & i=1,\ n;\ j=1,\ r_i x_i \\
P_l = B_1(y) & l=1,\ r
\end{cases} \quad\text{(2-7)}
$$

If (2-2), the fuzzy model's structure is defined as the number of classes and membership function of each universe of discourse of variables, parameter is the fuzzy relation R, the problem of recognition can be summarized as: Set sample data:

$\{x_i(t),\ y(t);\ i=1,\ \dots,\ n;\ t=1,\ \dots,\ N\}$, The destination of fuzzy model recognition is to define the structure of the model and the fuzzy relation R, here, N is the number of data samples. We choose to perform the model structure parameter recognition by mean of the previous described fuzzy correlation analysis method.

3. Algorithm

3.1 Recognition of fuzzy relation R

The core of the recognition job is to figure out the fuzzy relation R through the systemic I/O data. The R based on the reference fuzzy set is different in form and consideration from the R which is an element of the universe of discourse of direct connection. The membership function of the R, R $(s_1,\ \dots,\ s_n,\ s)$ Shows the relationship of reference sets $A_1s_1,\ \dots,\ A_ns_n,\ B_6,\ (s_1,\ \dots,\ s_n,\ s=1,\ r)$,

For the time data T, $x_i(t);\ y(t);\ i=1,$ n, the feasibility distribution of it can be figured out from (2-6),

$$
\begin{cases}
P_i(t) = [P_{i1}(t),\ P_{i2}(t),\cdots,\ P_{ir_i}(t)] \\
P(t) = [P_1(t);\ P_2(t),\cdots,\ P_r(t)]
\end{cases} \quad\text{(3-1)}
$$

Thus the sub-relation of this based on this time period

$$
\hat{R_i} = P_1(t) \times P_2(t) \times \cdots \times P_n(t) P(t) \quad\text{(3-2)}
$$

Here, "3" is the Cartesian product. For the syntheses algorithm, (3-2) can be spread as:

$$
\hat{R_i}(i_1,\ i_2,\cdots,\ i_n,\ i) = P_{1i_1}(t) * P_{2i_2}(t) * \cdots * P_{ni_n}(t) P_i(t)
$$
$$
i_1 = 1,\ r_1;\cdots i_n = 1,\ r_n;\ i=1,r \quad\text{(3-3)}
$$

When the max-min syntheses algorithm is adopted, simply replace the "3" with min in (3-2). The fuzzy overall relation when the number sample data is N,The union of the sub-relation produces:

3.2 Composition method of the membership function

Defining the membership function correctly is the essential part to use fuzzy set theory in solving practical problems, and sure is the foundation of building fuzzy model. Membership function is the rated description of the fuzzy concept. The method to define a membership function, should be a matter objective course, but individual knowledge and understanding of a certain fuzzy concept differs from each other, thus the subjectivity will be unavoidable when defining the membership function by mind according to the operator's experience. Meanwhile, the prior knowledge of the systemic variables' distribution is required, so, it's sometimes difficult to the model creator. Therefore, a virtual algorithm which can automatically output the membership function according the sample data is urgently needed.

General C----plot

Sample space

$X=\{x_1,\ x_2,\ \dots,\ x_n\},\ x_j=\{x_{j1},\ x_{j2},\ \dots,\ x_{jm}\}$ Is defined as dimensional vector, X for commonness, C-----plot is to

divide X into C category, i.e. the number of the nonvoid subset of X is C. $\{A_i\,|i=1,\ 2,\ \dots,\ c\}$ should satisfy:

$$
\bigcup_{i=1}^{c} A_i = X
$$
$$
A_i \bigcap A_j = \phi \qquad \forall i \neq j
$$

Such plot can be described as a c3n boolean matrix (C----plot matrix).

$$
A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_n \end{bmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ a_{c1} & a_{c2} & \cdots & a_{cn} \end{pmatrix}
$$

Where:

$$
a_{ij} = \begin{cases} 1 & X_j \in A_i \\ 0 & X_j \notin A_i \end{cases} \quad\text{(3-4)}
$$

Satisfy:

$$\sum_{j=1}^{n} a_{ij} = 1 \quad j = 1,2,\cdots, n \quad \text{———(3-5)}$$

$$\sum_{i=1}^{n} a_{ij} > 0 \quad i = 1,2,\cdots, c \quad \text{———(3-6)}$$

(3-5) shows that each sample x_j should belong to but only one category,(3-6) shows that each category Ai should have at least one sample.

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Shows that X $= \{x1,x2,x3,x4,x5\}$ is divided into categories C $=3$, they are $(x1,x5),(x2,x3),(x4)$. different C----partition matrix describes different partition.

In each category Ai, the vector Vi, the average of each characteristics of the samples, is the clustering center of Ai. Because the number of samples of Ai is $n_i = \sum_{j=1}^{n} a_{ij}$, Ai,

The vector sum of the samples of Ai is $\sum_{j=1}^{n} a_{ij} X_j$,

Wherefore clustering center is: $r_i = \sum_{j=1}^{n} a_{ij}/X_j \sum_{j=1}^{n} a_{ij}$.

can be described as:

$$V = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_c \end{bmatrix} = \begin{pmatrix} V_{11} & V_{12} & \cdots & \cdots & V_{1m} \\ V_{21} & V_{22} & \cdots & \cdots & V_{2m} \\ \bullet & \bullet & \cdots & \cdots & \bullet \\ \bullet & \bullet & \cdots & \cdots & \bullet \\ \bullet & \bullet & \cdots & \cdots & \bullet \\ V_{c1} & V_{c2} & \cdots & \cdots & V_{cm} \end{pmatrix}$$

If $X_j \in A_i$, then the distance from X_j to the clustering center V_1 is:

$$d_{ij} = \| X_j - V_i \| = (\sum_{k=1}^{\infty} (X_{jk} - V_{ik})^2)^{1/2},$$

The sum of squares of the distance of all the samples to the center is: $\sum_{i=1}^{n} a_{ij}(d_{ij})^2$.

In the partition, the sum of squares of all the samples to their respective clustering center is

$$J(A, V) = \sum_{i=1}^{n} \sum_{i=1}^{n} (a_{ij})^r \| X_j - V_i \|^2$$

Optimal C--plotting is obviously, when A of the

functional j (A, V) is minimized.

Fuzzy C----Partition

Extend the concept of general C---plot, And define fuzzy C---plot as follows:

Definition, sample space X $= \{x1,x2,\ldots,xn\}$,

$x_j = \{x_{j1}, x_{j2}, \ldots, x_{jm}\}$, a C3N fuzzy matrix:

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_c \end{bmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ a_{c1} & a_{c2} & \cdots & a_{c3} \end{pmatrix}$$

This is C---fuzzy plot matrix, if the following is satisfied:

$$\sum_{i=1}^{n} a_{ij} = 1 \quad j = 1,2,\cdots, n \quad \text{——(3-7)}$$

$$0 < \sum_{i=1}^{n} a_{ij} < n \quad i = 1,2,\cdots, c \quad \text{——(3-8)}$$

(3-7) shows that sum of each sample X_j which belongs to C fuzzy sub-category is 1.

(3-8) shows that each category Ai is not equal to ϕ and X, if ϕ and X are also sub-category, then (3-8) can be in the form of:

$$0 \leq \sum_{i=1}^{n} a_{ij} \leq n \quad i = 1,2,\cdots, c$$

Such plot is known as degraded fuzzy C---plot.

A C---fuzzy partition matrix present C---fuzzy partition of X and divides X into fuzzy subsets A_1, A_2, $\ldots A_c$, number of which is C. All the C---fuzzy partition composes C---fuzzy partition space. If degraded partition is included, it is known as degraded C---fuzzy partition space.

Supposing:

$$V_1 = \sum_{j=1}^{n} (a_{ij})^r X_j \Big/ \sum_{j=1}^{n} (a_{ij})^r$$

V1 is the clustering center of A1. The functionelle:

$$J(A, V) = \sum_{i=1}^{n} \sum_{j=1}^{n} (a_{ij})^r \| X_j - V_i \|^2$$

Where r/1 is a parameter, is a random norm of space Rn. Optimal C---fuzzy partition is to minimize J (A, V). (Parameter r is intended to level the degree of membership of Xj to A)

ISODATA method:

Dunn and Bezdek have proved that the optimization of the objective function J(A, V) of the

degraded C----fuzzy partition space is possible, the iterative algorithm Bezdek provided for the optimum solution of A when r/1 and $X_j \neq V_1$, it is called SODATA method. The steps as shown as follows:

（1） Set c: $2 \leq c \leq n$, and initial value $A^{(o)}$, (C----fuzzy partition matrix),

then iterate step by step, I=0, 1, 2...

（2） Calculate the class center matrix in step 1.

$$V^{(1)} = \begin{bmatrix} V_1^1 \\ V_2^1 \\ \vdots \\ V_c^1 \end{bmatrix} = \begin{pmatrix} V_{11}^1 & V_{12}^1 & \cdots & V_{1m}^1 \\ V_{21}^1 & V_{22}^1 & \cdots & V_{2m}^1 \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ \bullet & \bullet & \cdots & \bullet \\ V_{c1}^1 & V_{c2}^1 & \cdots & V_{cm}^1 \end{pmatrix}$$

$$V_i^1 = \sum_{j=1}^{n} (a_{ij}^{(1)})^r X_j \Big/ \sum_{j=1}^{n} (a_{ij}^{(1)})^r$$

（3） Modify $A^{(1)}$

$$a_{ij}^{1+1} = [\sum_{k=1}^{n} (\|x_j - V_i\|)/\|x_j - V_k\|^{1/(r-1)}]^{-1}$$

Compare $A^{(1)}$ and $A^{(1+1)}$ with the use of a matrix norm $\| * \|$, for a fixed $|\varepsilon| > 0$, if $\|A^{(1+1)} - A^{(1)}\| \leq \varepsilon$, then, stop the iteration, otherwise set to 1+1, then go to step 2.

In this method, r is usually set to 2, for a flexible alternation of the degree of membership of Xj to Ai. The selection of the initial value $A^{(o)}$ take effect in the rapidity of convergence, and the stand or fall of the final classification. The number of the classes also affects the final result.

When optimized C---partition is achieved, we can have the clustering center of each class.

$$V = \begin{bmatrix} V_1 \\ V_2 \\ \cdots \\ V_c \end{bmatrix}$$

We consider the center of the clustering has the greatest degree of membership to the class, i.e. $\mu_1 (V_1) = 1$, and on the midpoint of 2 adjacent clustering center, the fuzzy degree reaches its maximum value, i.e.

$\mu_i[(y_i + V_{i+1})/2] = 0.5$. Meanwhile, we define the membership functions as linear functions, thus to

construct the membership function of each class. Obviously, membership function of each class has a shape shown in figure 1, and the mathematic expression is:

$$\mu_i(x) = \begin{cases} (X - V_{i-1})/(V_i - V_{i-1}) & V_{i-1} \leq X \leq V_i \\ 0 & Other \\ (X - V_{i+1})/(V_i - V_{i+1}) & V_i \leq X \leq V_{i+1} \end{cases}$$
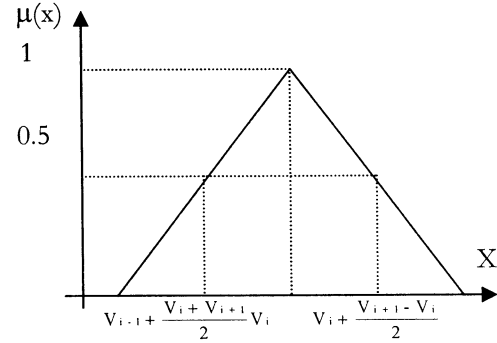
Figure 1.   Membership function

4.Summary

In the nearest 2 decades, the theory of systemic recognition and parameter estimation, the application of which is recognized as one of the most valuable approach to building the mathematical model of a controlled system where modeling approach itself is the precondition and foundation to perform system analysis, design, forecast, control and to make decisions, have made great progress and is updated to be a bouncing and important branch of automatic control theory.

References

[1].recognition using fuzzy set theory. Automation Transaction. May，1995Vol.17，No.3 pp.16-18

[2].Renzhong Zhang, The fuzzy controller design in online modification of control law.   Information and Control. 1996 Vol.24 ，No5 pp.27-30

[3].Chiu S. Fuzzy Logic for Control of Roll and Moment for a Flexible Wing Aircraft. IEEE Control Systems Magazine, 1991,11 pp. 42- 46

[4].Brisk. Procedure Control: Theory and Profit.12th IFAC, 1993 pp.332-335

[5].Bingyong Yang, Fuzzy model recognition and its applications, China environmental science press,1995 pp.228-230

Design of Autonomous Mobile Robot Action Selector based on a Fuzzy Artificial Immune Network

Dong-Je Lee
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
dongje01@bcline.com

Hong-Min Oh
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
hong1918@hanmail.net

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

Abstract

This paper addresses the low-level behavior of fuzzy control and the high-level behavior selector for Autonomous Mobile Robots (AMRs) based on a Fuzzy Artificial Immune Network. The sensing information that comes from ultrasonic sensors is the antigen it, and stimulates antibodies. There are many possible combinations of actions between action-patterns and external stimulation. The question is how to handle the situations to decide the proper action. We propose a fuzzy artificial immune network to solve the above problem. The computer simulation for an AMR action selector shows the validity of the proposed action selector.

Keywords : Immune network, Fuzzy Logic, Action Selector, Autonomous Mobile Robot

1. Introduction

Navigation algorithm for AMRs may be divided into two categories. The first one is the global path planning based on complete information about the whole environment, and the other one is the local path planning based on sensory information in uncertain environment. The global path planning is not suitable for navigation in complex and dynamically changing environments where unknown obstacles may be located on a priori planned path. Thus the sensor-based local path planning must be added to the global path planning in this situation.

Many local path plannings using the potential field [1], fuzzy system [2,3], and neural network [4] have been proposed. Fuzzy system can generate fuzzy rules by using expert information. In case of many inputs, however, a great deal of care and effort is required to obtain the rules. Neural network can't reach goal or may collide against obstacles in the environment without learning.

In recent years, researchers have focused on behavior-based path planning to acquire adaptive performance in the changing environments. A hierarchical fuzzy system of traditional behavior-based algorithms have been divided into three levels and have avoided moving obstacles by using inserted pre-prediction module and fuzzy balancer [5]. This method reduced the number of rules from 144 to 44, but it still has complex hierarchical structure. Another method of traditional behavior-based algorithms have used artificial immune networks imitating the human being's immune system, and the obstacle avoidance and goal approach behavior were built with artificial immune networks [6-8]. The final output was the sum of outputs of two behaviors with fixed weights; however, its performance was not continuous and it is difficult to select the adequate behavior in dynamically changing environments because of fixed weights.

This paper proposes a new action selector navigation algorithm that consists of 18 rules as the low-level fuzzy controller and an improved immune network as a supervisor. Because the improved fuzzy artificial immune network is simple, the immune network composition is easy. The fuzzy artificial immune network uses membership function to determine antibody easily. As a result, the composed antibody can express linguistic variables.

The stimulation and suppression chains among antibodies are similar to human being's immune system reacting the external antigen. We carried out some simulations to verify that our proposed method has stability and flexible features in diverse environments.

2. The human Being's Immune System

The basic structure of a human being's immune network consists of the T-lymph and B-lymph. There are approximately 10^7 of other kinds of B-lymph in a human body. As shown in Fig. 1, an antibody is Y-shaped and has a paratope and an ideotope, An artificial immune network was suggested by N. K. Jerne [9].
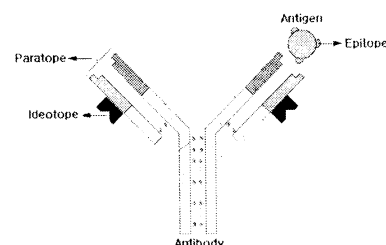
Fig. 1. Structure of antibody and antigen

3. Entire Action Control Scheme for Autonomous Mobile Robot

In order to acquire information about unknown environments, 9 ultrasonic sensors are mounted on the AMR. These ultrasonic sensors are divided into three groups to detect obstacles from the left, front and right locations, as shown in Fig. 2.
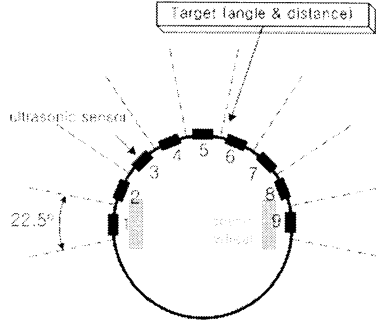
Fig. 2. Unknown environments sensing of AMR

The AMR is driven with two wheels. The input signals to the low-level fuzzy logic navigation algorithm are the minimum distance and angle between the robot and obstacles as well as the heading angle and distance between the robot and specified target. The low-level fuzzy logic behaviors are the left-hand-law action, target steering action and obstacle avoidance with goal tracking. The input data to high-level behavior selector by the fuzzy artificial immune network are the numbers of detection for the right, front and left directions as well as the target approach rate, target approach trend rate and target distance.
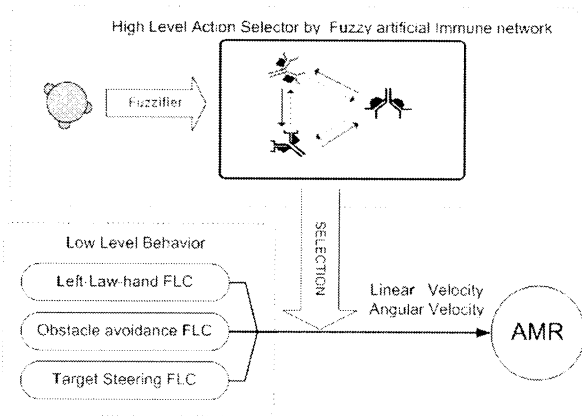
Fig. 3. Entire action control scheme

4. Low-Level Behaviors

4.1 Membership Functions of Inputs

Unit of distance is pixel on simulation screen. Unit of angle is degree.
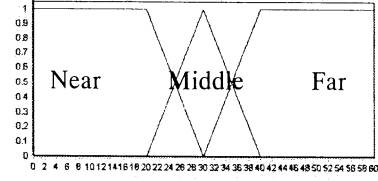
Fig. 4. Membership functions of obstacle distance input
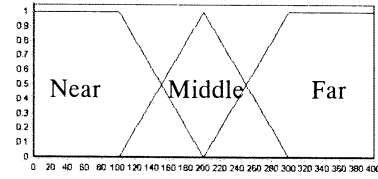
Fig. 5. Membership functions of target distance input
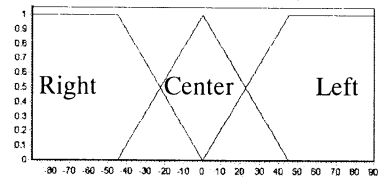
Fig. 6. Membership functions of obstacle & target direction input

4.2 Membership Functions of Outputs
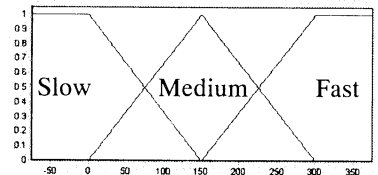
Fig. 7. Membership functions of linear velocity output
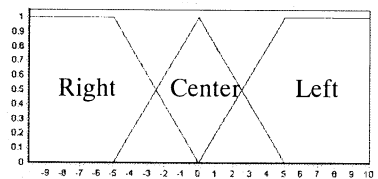
Fig. 8. Membership functions of angular velocity output

4.3 Left-Hand-Law by Fuzzy Logic Control

This behavior is only for avoiding obstacle. Rules are composed of the distance and direction of the obstacle. The linear velocity and angular velocity output rules are shown in Tables 1 and 2, respectively.

Table 1. Rules of linear velocity

Direction / Distance	Right	Center	Left
Near	Medium	Slow	Medium
Middle	Medium	Medium	Medium
Far	Fast	Fast	Fast

Table 2. Rules of angular velocity

Direction / Distance	Right	Center	Left
Near	Left	Right	Right
Middle	Center	Right	Center
Far	Left	Left	Left

4.4 Obstacle Avoidance by Fuzzy Logic Control

This behavior considers both the obstacle avoidance and target tracking. Rules consist of the distance and direction of the obstacle and goal. The linear velocity and angular velocity output rules are shown in Tables 3 and 4, respectively.

Table 3. Rules of linear velocity

Direction / Distance	Right	Center			Left
		Target direction			
		Right	Center	Left	
Near	Slow	Slow	Slow	Slow	Slow
Middle	Med	Med	Med	Med	Med
Far	Fast	Fast	Fast	Fast	Fast

Table 4. Rules of angular velocity

Direction / Distance	Right	Center			Left
		Target direction			
		Right	Center	Left	
Near	Left	Left	Left	Right	Right
Middle	Left	Right	Center	Left	Right
Far	Left	Right	Center	Left	Right

4.5 Target Steering by Fuzzy Logic Control

This is only about the target tracking behavior. Rules use the distance and direction of the target. The linear velocity and angular velocity output rules are shown in Tables 5 and 6, respectively.

Table 5. Rules of linear velocity

Direction / Distance	Right	Center	Left
Near	Fast	Fast	Fast
Middle	Med	Fast	Med
Far	Med	Med	Med

Table 6. Rules of angular velocity

Direction / Distance	Right	Center	Left
Near	Right	Center	Left
Middle	Right	Center	Left
Far	Right	Center	Left

5. High-Level Behaviors Control

Table 7 depicts definition of an antigen and antibody. The input about obstacle information has values between 0 and 10 that is the detected number of times during the latest 10 sampling times. Current Approach (CA) of inputs about target information expresses the difference between the present position and past position. Approach Trend (AT) is the value that accumulates data of current approach during the latest 10 sampling time. Current Distance (CD) denotes the distance between target and AMR. For antibody's details, Obstacle Detected Count have *Few, Middle, Many* membership functions. Target's CA and AT have *Negative, Zero, Positive* membership functions. CD have *Near, Middle, Far* membership functions.

Table 7. The definition of an antigen and antibody structure.

Obstacle Detected Count			Target		
Right	Front	Left	Current Approach	Approach Trend	Current Distance

5.1 Immune Membership Functions of Inputs
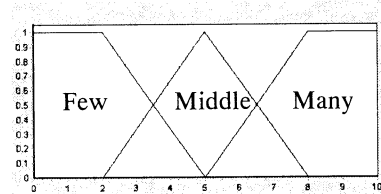
Fig. 9. Membership functions of obstacle count input
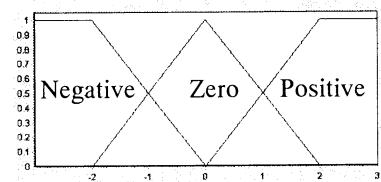
Fig. 10. Membership functions of CA input
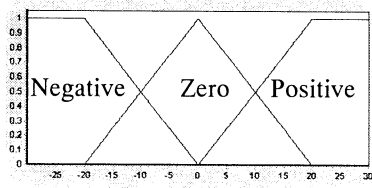
Fig. 11. Membership functions of AT input
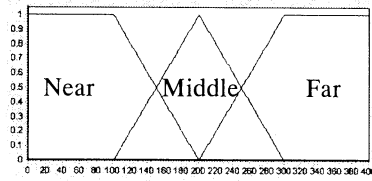
Fig. 12. Membership functions of CD input

5.2 Structure of Fuzzy Artificial Immune Network
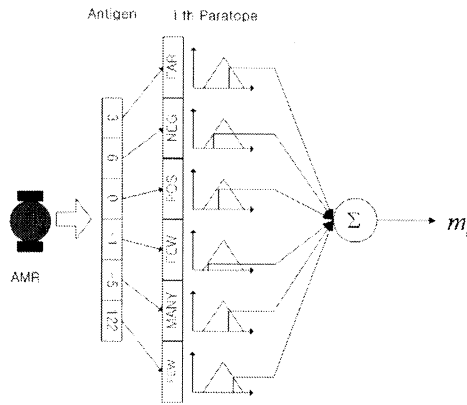
Fig. 13. Fuzzifier for antigen input data

The structure of action selector by the fuzzy artificial immune network is shown in Fig. 13. We obtain the activation rate between antigen and i-th antibody through fuzzified external inputs. Table 8 depicts description of the paratope.

Table 8. The setting of the paratope

	Rule of Paratope					
LHL	Many	Many	Many	Neg	Neg	Far
OB	Few	Many	Few	#	Pos	Far
TS	Few	Few	Few	Pos	#	Near

LHL : Left-Hand-Law behavior
OB : Obstacle Avoidance behavior
TS : Target Steering behavior
: Don't Care

The relation of stimulation and suppression between antibodies is shown in Table 9.

Table 9. The setting of stimulation & suppression matrix

	LHL	OB	TS
LHL	0	1	-1
OB	1	0	-1
TS	-1	-1	0

5.3 Concentration Equation of Antibody

After the initial concentration value is 0.1, new behavior is selected when the concentration reaches the limit 0.8. The following equations (1) and (2) present the concentration on the i-th antibody[6].

$$\frac{da(k)_i}{dt} = \left\{ \frac{\sum_{j=1}^{N} w_{i,j} \cdot a(k-1)_j}{N} + m_i \right\} \cdot a(k-1)_i \quad (1)$$

$$a(k)_i = 1/\exp\left\{ 0.5 - \left(a(k-1) + \frac{da(k)_i}{dt} \right) \right\} \quad (2)$$

6. Results of Computer Simulation

6.1 The Condition of Simulation

The environment for the AMR's consists of the obstacles and goals. The AMR radius is $10[pixel]$. The AMR's maximum speed is 3 $[pixel/sample\ time]$. The AMR's sensing distance is 6 times longer than a robot's radius.

6.2 Result of the Action Selector based on IF-then Rule
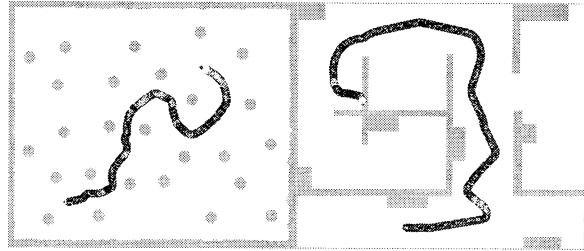
Fig. 14. Result of the action selector by if-then rule with target tracking property
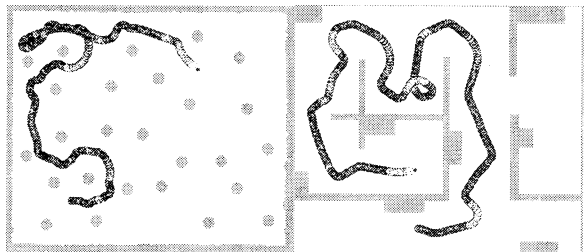
Fig. 15. Result of the action selector by if-then rule with left-hand-law property

Figs. 14 and 15 show the action selector by the if-then rule using the same low-level controller. We give more weight to target tracking than obstacle avoidance in Fig. 14. It is opposite with above. Fig. 14 depicts target tracking feature. Fig. 15 depicts obstacle avoidance feature. As shown in the above figures, show disadvantage that do not reach target or unnecessary action.

6.3 Result of the Action Selector based on a Fuzzy Artificial Immune Network
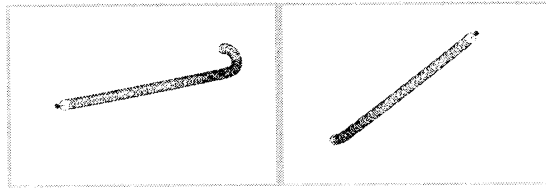
Fig. 16. Result of the proposed algorithm in the environment without obstacle
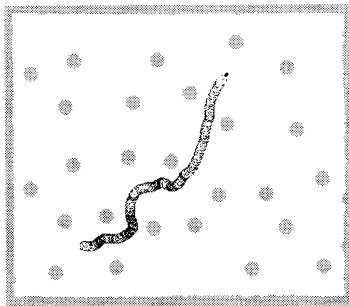
Fig. 17. Result of the proposed algorithm in the environment with circular obstacles
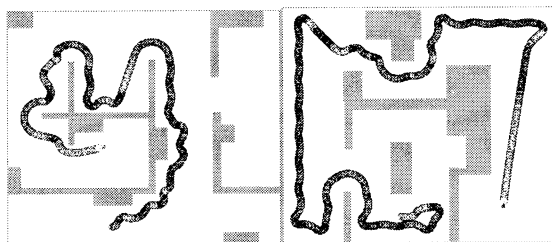
Fig. 18. Result of the proposed algorithm in the indoor environment

To demonstrate the effectiveness and robustness of the proposed method, AMR conducts target tracking without action such as the left-hand-law in the environment without obstacles as shown in Fig. 16. AMR also can have the obstacle avoidance and target tracking action in the environment with circular obstacles shown in Fig. 17. In the indoor environment in Fig. 18, AMR can escape from the local obstacle environment without target tracking, and after escape it resumes target tracking to reach the final target.

7. Conclusion

In this paper, we use the fuzzy artificial immune network to realize the action selector for AMR navigation. The proposed algorithm has much better adaptability than the if-then rule. Moreover, the internal antibody can be expressed as the linguistic variable using fuzzifier method and make use of real number inputs instead of binary inputs in traditional immune networks. Therefore, it is possible to utilize expert acknowledge due to the linguistic expression. Our fuzzy artificial immune network can be composed more conveniently than the conventional immune networks.

Reference

[1] O. Khatib, " Real-time obstacle avoidance for manipulators and mobile robots, " *Proc. IEEE International Conference on Robotics and Automation*, pp. 500-505, 1985.

[2] H. R. Beom and H. S. Cho, "A sensor-based obstacle avoidance controller for a mobile robot using fuzzy logic and neural network," *Proc. IEEE International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1470-1475, 1992.

[3] A. Ramirez-Serrano and M. Boumedine, "Real-time navigation in unknown environments using fuzzy logic and ultrasonic sensing," *Proc. IEEE International Symposium on Intelligent Control*, pp. 26-30, 1996.

[4] K. P. Prabir and K. Asim, " Mobile robot navigation using a neural net," *Proc. IEEE International Conference on Robotics and Automation*, pp. 1503-1508, 1996.

[5] T. Aoki, M. Matsuno, T. Suzuki and S. Okuma, "Motion planning for multiple obstacles avoidance of autonomous mobile robot using hierarchical fuzzy rules," *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 265-271, 1994.

[6] Akio Ishiguro, Yuji Watanabe and Yoshiki Uchikawa, " An immunological approach to dynamic behavior control for autonomous mobile robots," *Proc. IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 495-500, 1995.

[7] Akio Ishiguro, Yuji Watanabe and Toshiyuki Kondo, " Proposal of decentralized consensus-making mechanisms based on immune system, " *Proc. International Symposium on Artificial Life and Robotics*, pp.122-127, 1996.

[8] Akio Ishiguro, Toshiyuki Kondo and Yuji Watanabe, "An immunological approach to dynamic behavior arbitration for autonomous mobile robots," *Proc. International Symposium on Artificial Life and Robotics*, pp. 132-137, 1996.

[9] N. K. Jerne, "The immune system," *Scientific American*, vol. 229, no. 1, pp. 52-60, 1973.

STUDY ON A NEW AND EFFECTIVE FUZZY PID SHIP AUTOPILOT

Minh-Duc LE

Vietnam Shipbuilding Industry Corporation (VINASHIN),
109 Quan Thanh, Ba dinh, Hanoi, Vietnam,
le_minh_duc@yahoo.com

Lan-Anh NGUYEN

Shipbuilding Science and Technology (ShipSciTech),
80B Tran Hung Dao, Hanoi, Vietnam

Abstract

Ship Autopilots are usually designed based on the PD and PID controllers because of simplicity, reliability and easy to construct. However their performance in various environmental conditions is not as good as desired. This disadvantage can be overcome by adjusting works or constructing adaptive controllers. But those methods are complex and not easy to do.

This paper presents a new method for constructing a Ship Autopilot based on the combination of Fuzzy Logic Control (FLC) and Linear Control Theory (PID control). The new Ship Autopilot has the advantages of both the PID and FLC control methodologies: easy to construct, and optimal control laws can be established based on ship masters' knowledge. Therefore, the new ship autopilot can be well adapted with parameter variations and strong environment effects. Simulation using MATLAB software for a ship with real parameters shows high effectiveness of the Fuzzy PID autopilot in course keeping and course changing manoeuvres in comparison with the ordinary PID ship autopilots.

Keywords: PID control, Linear control, Sliding Mode Control Methodology, Fuzzy Logic Control, Ship Autopilots, Ship Steering Dynamics

1 Introduction

In the last twenty years, together with the drastic development of micro-electronics and control theory, several new and effective methods have been proposed and developed for designing Ship Autopilots [1], [2], [3], [4], [5]. Ship Autopilots designed based on the PD and PID controllers are simple, reliability and easy to construct, however their performance in various environmental conditions is not as good as desired. Therefore, Ship Autopilots with PD or PID controllers are usually required aids from operators to adjust controllers' parameters corresponding to navigating conditions. This adjusting work is called "weather adjustment" and is an undesirable work for the operators. Parameter adjustment is a difficult task and if carried out not properly it can cause damages to machines and unnecessary fuel consumptions. Another more effective method is to construct adaptive controllers based on ship models. This method can be used for various types of ship described by nonlinear mathematical models. But on the other hand, it also has several disadvantages, one of these is its completeness since the range of parameters and environmental disturbances is usually very large.

A new method for constructing Ship Autopilots based on Variable Structure Controllers (VSC) or the so called Sliding Mode Controllers (SMC) was proposed by Papoulias and Healey. Using this method one can construct high quality Ship Autopilots that is not much influenced by variations of parameters and disturbances. Although the traditional Sliding Mode Controllers have several advantages, they have also a very weak point that is the saturation in the sliding surface. Working in this state, the controllers have to operate in rather high frequency that causes wasting energy largely and reduces the quality of the controllers [6], [7], [8].

Recently, a tendency that is of much interested is to combine actively various methodologies for exploiting the advantages of each one [6], [9], [10]. One of the most effective solutions is combining Linear Control Theory and Fuzzy Logic Control (FLC). This paper presents a new method to rise the quality of the Ship Autopilots, that is the Fuzzy PID Autopilots (FPID Autopilots) based on the combination of FLC and PID Autopilot. The new Ship Autopilot has the advantages of both the FLC and PID control methodologies. For example, a system of optimal control laws can be established based on ship operator experts' knowledge, so that the designed Ship Autopilot has behaviors similar to that operated by human being. Moreover, one notable advantage of the new Ship Autopilot is that it can be adapted with complicated ship steering systems with parameter variations and strong environmental effects.

In the first part of this paper, ship hydrodynamics and mathematical models used for expressing ship motions in steering maneuvers are presented. The second part of the paper gives summary of PID Ship Autopilot, FLC and their combination to construct the FPID Ship Autopilot. The new method has been applied to construct a FPID Ship Autopilot for a ship with real parameters. To verify effectiveness of the FPID Ship Autopilot, simulation of several ship steering maneuvers has been carried out with MATLAB software. Results of this simulation is presented in comparison with the ordinary PID Ship Autopilot in the third part of this paper. And finally, some main conclusions are drawn and future research directions are given.

2 The Mathematical Models expressing ship steering dynamics

2.1 General structure of a ship automatic steering system

The general structure of ship automatic steering system [11] is shown in Fig. 1.
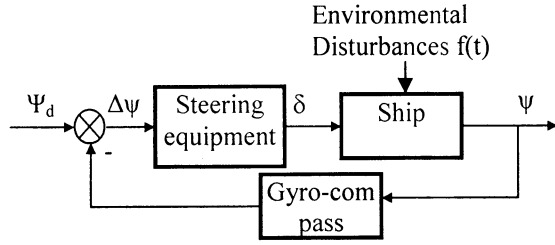
Fig. 1: General structure of ship steering systems

The system consists of:
- A steering equipment (steering machine): used to produce necessary forces for rotating rudder an required angle δ in an interval and keep the rudder at a fixed angle while ship is moving.
- Ship: control object.
- Gyrocompass: used to find the real ship heading angle ψ. This angle is compared with desired heading angle Ψ_d to calculate control error of heading:

$$\Delta\psi = \psi_d - \psi \qquad (1)$$

Based on this error, steering machines generates control signal to deflect rudder into a suitable angle to reduce error of heading and keep the ship in a predetermined direction.

2.2 Mathematical description

Corresponding to the general structure of a steering system shown in Figure 1, steering equipment can be divided into a controller and a rudder angle regulator (that is a amplifier with boundary limitation). The ship is usually expressed by a transmission function (between heading angle ψ and rudder deflection angle δ) for both course-stable and course-unstable cases are expressed as following (plus for course-stable case and minus for course-unstable case):

$$W_\delta(p) = \frac{\psi(p)}{\delta(p)} = \frac{k_c(1+\tau_1 p)}{p(T_2 p^2 + T_1 p \pm 1)} \qquad \text{(3-order)} \quad (2)$$

$$W_\delta(p) = \frac{\psi(p)}{\delta(p)} = \frac{k_c}{p(T_1 p \pm 1)} \qquad \text{(2-order)} \quad (3)$$

Then, block diagram of a simple autopilot can be drawn as shown in Figure 2.
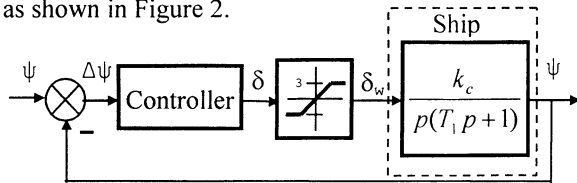
Fig. 2: Block diagram of a simple autopilot

2.3 Control method based on PID controller

In this method, rudder deflection angle $\delta(t)$ is

controlled by PID control law based on error of heading angle $e(t) = \psi_r(t) - \psi(t)$, here ψ_r reference (desired) heading angle, as expressed by following equation:

$$\delta(t) = K_p e(t) + K_d \frac{de(t)}{dt} + K_i \int e(t) dt \qquad (4)$$

, and the transmission function is as follows:

$$w_c(p) = \frac{\delta(p)}{e(p)} = K_p + K_d p + \frac{K_i}{p} \qquad (5)$$

Parameters K_p, K_i, K_d of the PID controller can be estimated using Routh-Hurwitz methodology [12]. Usually, only 2 equations can be established, where as the number of parameters is 3, then 1 parameter is chosen to satisfy some criteria and the others then calculated accordingly. In the case of 2-order system, the PID parameters can be chosen to satisfy control quality criteria such as overshoot, time of stable and so on. With a chosen K_i, K_p, K_d are calculated by:

$$K_p = \frac{-\sin(\beta+\varphi)}{\left|w_p(s_1)w_c(s_1)\right|\sin\beta} - \frac{2K_i\cos\beta}{\left|s_1\right|} \qquad (6)$$

$$K_d = \frac{\sin(\varphi)}{\left|s_1\right|\left|w_p(s_1)w_c(s_1)\right|\sin\beta} + \frac{K_i}{\left|s_1\right|^2} \qquad (7)$$

, here s_1 is the pole of closed-loop control system, β and φ are parameters defined by:

$$s_1 = \left|s_1\right|e^{j\beta} \qquad (8)$$

$$w_p(s_1)w_c(s_1) = \left|w_p(s_1)w_c(s_1)\right|e^{j\psi} \qquad (9)$$

2.4 The FPID controller

Fuzzy logic controllers (FLC) have good characteristics on the zones with large error of control, where the controllers can produce quick dynamic responses due to non-linear characteristics. Besides, when the controllers work near stable points, the role of FLC is not sufficient, and a PID controller may have better effectiveness.
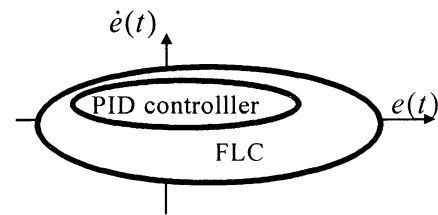
Fig. 3: FLC and PID controllers' effective zones

Changing from FLC to PID and vise versa is done by following simple laws:

if $\left|e(t)\right|$ **is PB and** $\left|\dot{e}(t)\right|$ **PB then u is** FLC $\qquad (10)$

if $\left|e(t)\right|$ **is PS and** $\left|\dot{e}(t)\right|$ **PS then u is** PID $\qquad (11)$

where PS, PB are condition expressions. This can be applied for cases when several PID controllers are used

and each of them has a pre-determined working zone. In these cases, changing laws has the following form:

Ru(i): **if** ER **is** E^p **and** CER **is** CE^q **then**

$$u_i = K_{Pi}e + K_{Ii}\int_0^t e(\tau)d\tau + K_{Di}\frac{de}{dt} \qquad (12)$$

$i = 1, 2, ..., n$

, where u_i is the working results of the ith PID controller. If define:

$$\sigma_i(t) = \frac{\left(\mu_i(e(t)) \times \mu_i(\dot{e}(t))\right)}{\sum_{i=1}^n \mu_i(e(t)) \times \mu_i(\dot{e}(t))} \qquad (13)$$

with $\mu_i(.)$ is the membership functions, then results of the FPID controller according to the system of laws (12) can be drawn as follows:

$$u(t) = K_{PN}e + K_{IN}\int_0^t e\,dt + K_{DN}\frac{de}{dt} \qquad (14)$$

$$\begin{cases} K_{PN} = \left(\sum_{i=1}^n \sigma_i(t)K_{Pi}\right) \\ K_{DN} = \left(\sum_{i=1}^n \sigma_i(t)K_{Di}\right) \\ K_{IN} = \left(\sum_{i=1}^n \sigma_i(t)K_{Ii}\right) \end{cases} \qquad (15)$$

3 Simulation results and evaluation

The FPID controller was designed for a 16.500 T named Kabec (a Russian flag ship) with following parameters:

$$k_c = (0.0311 \div 0.057)\sec^{-1}; T_1 = (14 \div 19)\sec \qquad (16)$$

Parameters of the PID controller are shown in Table 1.

Table 1: Parameters of the PID autopilot for the Kabec

K_P	K_I	K_D
90	40	400
150	55	380
205	75	250
98	95	180
335	125	130

If inputs and output are defined as follows:

$$\begin{cases} |e(t)| \doteq \{VPS, PS, PM, PB, VPB\} \\ |\dot{e}(t)| \doteq \{PS, PM, PB\} \\ u(PID_i) \doteq \{PID_1, PID_2, PID_3, PID_4, PID_5\} \end{cases} \qquad (17)$$

, and the de-fuzzification laws are chosen as shown in Table 2.

Table 2: De-fuzzification laws for FPID autopilot

$\frac{|e(t)|}{|\dot{e}(t)|}$	VPS	PS	PM	PB	VPB
PS	PID_5	PID_4	PID_3	PID_2	PID_1
PM	PID_5	PID_4	PID_3	PID_2	PID_1
PB	PID_5	PID_4	PID_3	PID_2	PID_1

Membership functions for $|e(t)|$, $|\dot{e}(t)|$ were chosen using Gaussian distribution form in [0, 1] interval as shown in Figures 4a, 4b.
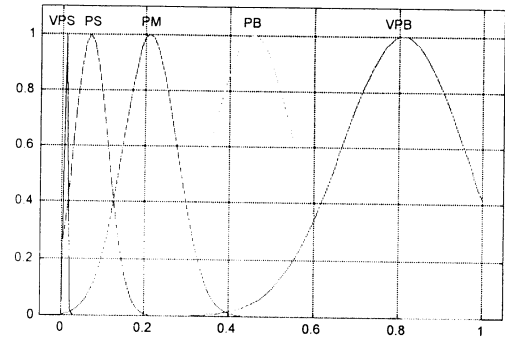
Fig. 4a: Membership function for heading angle error
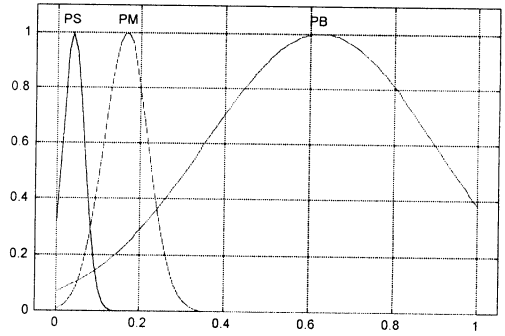
Fig. 4b: Membership function for ship turning rate

MATLAB simulation results of the PID autopilot and FPID autopilot are shown in Figures 5a, 5b, 5c.
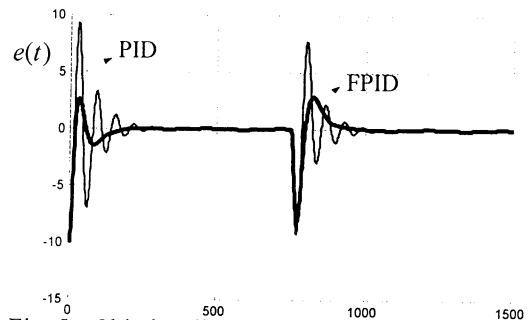
Fig. 5a: Ship heading error according to PID and FPID autopilots
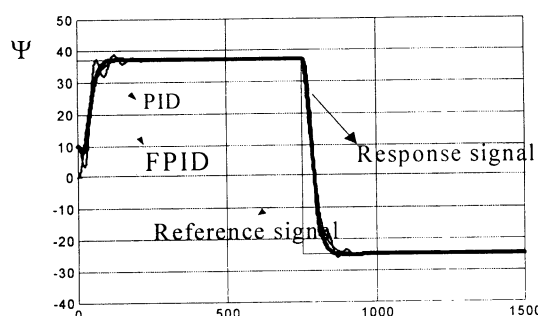
Fig. 5b: Ship heading angle according to PID
and FPID autopilots
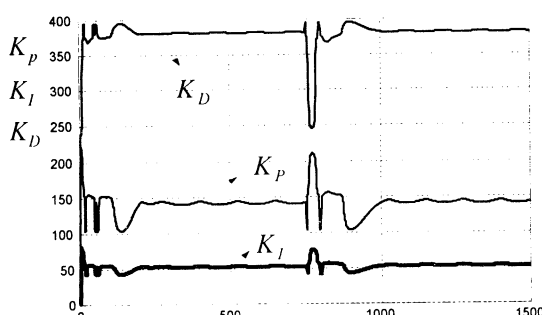
Fig. 5c: PID and FPID autopilots' parameters
during maneuvers

It is clearly seen that, in comparison with the PID
autopilot, the new FPID autopilot has several important
features:

- In general ship heading error (or deflection of ship
heading angle from reference values) of FPID autopilot
is smaller than that of PID autopilot;

- Times of over-correction and maximum values of
overshoot of heading angle for FPID autopilot are much
more smaller than that of the PID autopilot;

4 Conclusions and Future works

A new method for constructing Ship Autopilots based
on the combination of Fuzzy Logic Control (FLC) and
Linear Control Theory (PID control) has been presented.
The new Ship Autopilot has the advantages of both the
PID and FLC control methodologies: easy to construct,
and optimal control laws can be established based on
ship masters' knowledge. Simulation using MATLAB
software for a ship with real parameters shows that the
FPID autopilot has several important features in
comparison with the corresponding PID autopilot. The
new autopilot is much more effective than PID autopilot
in course keeping and course changing manoeuvres.

However, in order to be able to use the FPID autopilot
in practice, several problems should be solved, among
them are: what is most suitable form for membership
functions of the FPID controller, how to automatically
design the membership functions and de-fuzzification
laws for the autopilot, what criteria should be chosen for
the optimal autopilot, consideration of real aspects of
the autopilot, and so on.

References

[1] C. G. Kallstrom, "Identification and adaptive control
applied to ship steering", PhD thesis, Dept. of
Automatic Control, Lund Institute of Technology,
Sweden, 1970.

[2] C. G. Kallstrom, K. J. Astrom, N. E. Thorell, et. al.,
"Adaptive Autopilots for Tankers", Automatica, Vol.
15, 241-254, 1979.

[3] C. G. Kallstrom, K. J. Astrom, "Experience of
System Identification Applied to Ship Steering",
Automatica, Vol. 17 (1), 187-198, 1981.

[4] M. D Le, "Online Estimation of Ship Steering
Dynamics and Its Application in Designing An
Optimal Autopilot", Proc. of IFAC Computer Aided
Control System Design, CACSD2000, Vol. 1, 7-12,
2000.

[4] D. H. Nguyen, M. D. Le and K. Ohtsu, "Ship's
Optimal Autopilot with a Multivariate
Auto-Regressive eXogenous Model", Proceedings of
the 11th IFAC Workshop on Control Application of
Optimization (CAO2000), Vol. 2, 2000.

[5] D. H. Nguyen and M. D. Le, "A Challenge to
Advanced Autopilot Systems for Ships",
Proceedings of the 4th Vietnam Conference on
Automation (VICA4), Vol. 1, 219-225, 2000.

[6] T. I. Fossen, Guidance and Control of Ocean
Vehicles, John Wiley&Sons, Chichester, 1994.

[7] D. R. Yoerger and J. J. E Slotine, "Nonlinear
Trajectory Control of Autonomous Underwater
Vehicles using Sliding Methodology", Proceedings
of the ROV'84 Conference, Vol. 1, 245-251, 1984.

[8] D. R. Yoerger and J. J. E Slotine, "Adaptive Sliding
Control of an Experimental Underwater Vehicle",
Proceedings of IEEE International Conference on
Robotics and Automation, Vol. 1, 2746-2751, 1991.

[9] K. Agata, Modern Control Engineering, Kluwer
Academic Publishers group, 1994

[10] K. J. Astrom, B. Wittenmark, Adaptive control,
Academic Press, 1989.

[11] Astrom, K. J. and Kallstrom, C. G: Identification of
Ship Steering Dynamics, Automatica, Vol.12, No.1,
9-22, 1976.

[12] V. N. Afanasiev, V. Konmagorovskii and V. R.
Nosov, Mathematical theory of control systems
design, Kluwer Academic Publishers group, 1995.

Application of Neuro-Fuzzy System to Control a Mobile Vehicle

Masanori Sugisaka [1]   Fengzhi Dai [2]
Department of Electrical and Electronic Engineering
Oita University, Oita, Japan
[1] msugi@cc.oita-u.ac.jp   [2] daifz@cc.oita-u.ac.jp

Hidenori Kimura [3]
The Institute of Physical and Chemical
Research at Nagoya, Japan
[3] kimura@crux.t.u-tokyo.ac.jp

Abstract

This paper presents a practical control method for the experimental mobile vehicle. By merging the advantages of the neural network, adaptive algorithm and fuzzy control, the adaptive fuzzy control based on neural network is proposed. This adaptive fuzzy control system can deal with a large amount of training data by the neural network, from these data produce more reasonable fuzzy rules by the adaptive (clustering) algorithm, at last control the object by the fuzzy control. It is not the simple combination of the three methods, but merging them into one control system. Experiments and some future considerations are also given.

Keywords: hybrid system, neural network, clustering algorithm, fuzzy control, adaptive fuzzy control, mobile vehicle

1. Introduction

Traditional control systems are always based on mathematical models, but in many cases, the mathematical models of the control process may not exist, or may be too difficult to get them.

Neural network (NN) control is different from the conventional methods. It can deal with a large amount of training data, and from which the relationships among them can be found, although unrelated information and noise are in them (Wasserman [1]).

Also, in a system, the values of variables may be within a range of states. Sometimes the transition from one state to another is hard to define. The way to solve it is to make the states "fuzzy": allow them to change gradually from one state to another (Goebel [2]).

Each control method has its advantages and disadvantages. The paper wants to merge them, so that the advantages will be adopted while the weakness eliminated (Abe [3]). It is a mechanism, by which the control system can be utilized from the expert knowledge, to solve a special problem.

1.1 Hybrid systems

When more than one problem-solving technique (NNs, fuzzy logic, expert systems, etc.) are used in order to solve a problem, they are called hybrid systems. There are three main structures for them to be connected into one hybrid system shown in Gray [4]:

the sequential, auxiliary and embedded hybrid.

We have to point out that: although hybrid systems are intrinsically better, which allow for the synergistic combination of two techniques with more strength and less weaknesses than either alone, they have more chances for misuse than single technique. These should be considered when construct a new hybrid system.

1.2 AFCNN hybrid system

The data-driven learning technique (NNs) and knowledge-driven technique (the fuzzy system) will be considered in this paper. The integration of NN and the fuzzy control is called the neuro-fuzzy hybrid system. In fact, it is based on using NN to simulate a fuzzy system, or to adjust its membership functions.

The experimental object is an intelligent mobile vehicle (MV). There are many methods presented to control MV or robots. Watanabe et al [5] presented the fuzzy behavior-based system for a mobile robot. It is convenient to add new behavior functions into the system, but for the stored functions, they are difficult to be adjusted based on the changeable environment. Wang et al [6] developed a fuzzy logic Kalman filter estimation method for a two-wheel steerable vehicle. This method merged the fuzzy control and Kalman filter, but it is not an intelligent control system.

Adaptive fuzzy control based on neural network (AFCNN) is one of the neuro-fuzzy hybrids, which has embedded structure. It is capable of extracting and optimizing fuzzy rules and sets from the input-output training data. The clustering methods are used to produce fuzzy rules. And by clustering techniques to mine the important input-output relationship from the input- output data pairs to form and optimize the fuzzy rules and sets is the main thought of the paper.

In this paper, section 2 gives the general configuration of the experimental mobile vehicle. In section 3, AFCNN system is described. Section 4 gives the experimental results and analysis. In section 5, conclusions are drawn. The object of the paper is to present an effective and practical control method, which can be used to obtain the reasonable fuzzy control rules adaptively by NN.

2. Structure of the mobile vehicle

Fig.1 gives the photo of the experimental MV. It is 1.5m×0.8m×0.4m and 150kg. The MV system is

comprised of a rigid metal frame to support the equipment, a CCD camera and digitizer, I/O systems and power supplies that enable completely autonomous operation, and a computer is the control center.
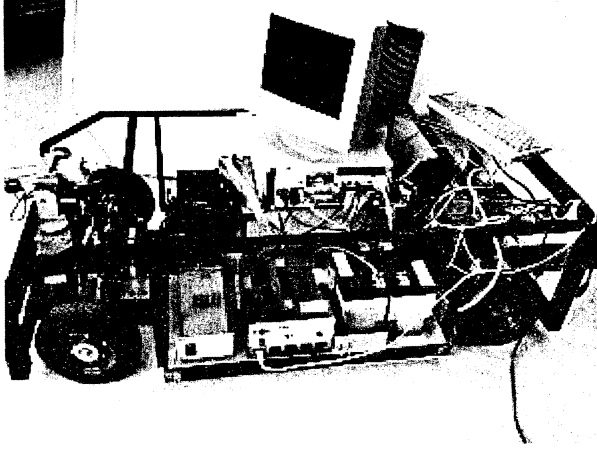
Fig.1　Mechanical configuration of MV

From the motion control perspective, MV has two degrees of freedom: velocity and angular velocity (by the driving and steering wheels, respectively). The experimental MV has one stepping motor to drive MV rear wheels to move forward or backward, as well as one stepping motor to steer the front wheels to rotate left or right. An encoder is used to detect the real rotation angle of the front wheels. The sensor is a CCD camera. The goal of the control is to make MV move on the road tracking the guideline smoothly.

3. Structure of AFCNN

By using the fuzzy logic, the inference from experience can be obtained. And NN is used to cluster the training data by the adaptive algorithm, so that the fuzzy rules and sets can be optimized. That is the principle of the AFCNN system. First, a fuzzy rule producer will be considered.

3.1 Fuzzy rule producer

The general fuzzy model is shown in Eq.(1). Fuzzy rules for a two-input one output system are

$$\begin{cases} \text{Rule 1 : If } x \text{ is } A_1 \text{ and } y \text{ is } B_1 \text{, then } z \text{ is } C_1, \\ \text{Rule 2 : If } x \text{ is } A_2 \text{ and } y \text{ is } B_2 \text{, then } z \text{ is } C_2, \\ \quad \cdots \quad\quad \cdots \quad\quad \cdots \\ \text{Rule m : If } x \text{ is } A_m \text{ and } y \text{ is } B_m \text{, then } z \text{ is } C_m. \end{cases} \quad (1)$$

Here x, y and z are fuzzy variables. A_i, B_i ($i = 1, 2, ...,$ m) are fuzzy values that belong to {NB, NM, NS, ZE, PS, PM, PB}, which can be described by membership functions. NB, ZE, PS are fuzzy terms that representing negative big, zero, positive small, respectively. If the output C_i is also a fuzzy value, it is type 1. If C_i is a

singleton constituent, it is type 2.

And if the output C_i is the function of the input data, it is type 3. Takagi and Sugeno [7] fuzzy algorithm can handle this kind of fuzzy models and then accurate output from fuzzy rules can be obtained.

3.2 AFCNN

Section 3.1 shows that from the input-output sets, the complicated relationship between inputs and outputs can be simulated by fuzzy rules. Now the problem is: faced to a large amount of the given input-output training sets, how to get the fuzzy rules adaptively. AFCNN can solve this problem, and is given in Fig.2.
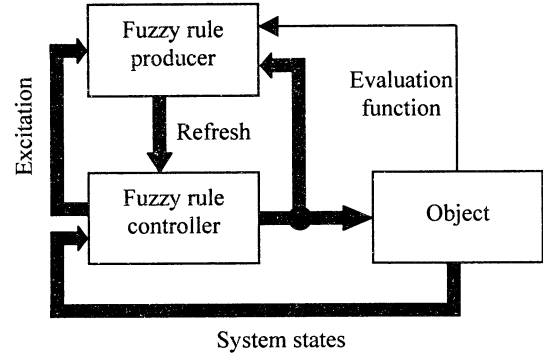
Fig. 2　Structure of AFCNN

To form and adjust fuzzy rules, first, the centers of membership functions are founded by dividing each input and output linguistic variable into predetermined number of fuzzy values. The widths of fuzzy sets are initially determined by nearest neighbor or k-means heuristics (Ojala [8]). After that, the fuzzy rules are found by feeding training data set through AFCNN and deciding which fuzzy rules are the most important. Some similar consequences can be combined into one. Finally, the centers and widths of the fuzzy sets are turned using a gradient descent algorithm.

For training the data, it starts with a reasonable number of clusters and shrink the number by certain algorithms (e. g., the following competitive learning adaptive vector quantization (CL-AVQ) algorithm).

Thus AFCNN is divided into the following four steps:

(1) Define the fuzzy universe of discourse for both input and output variables (the number and centers for fuzzy rules). For example, input fuzzy sets are {I_1, I_2,..., I_r}, outputs are {O_1, O_2,..., O_s}. I_i (i = 1, ..., r) and O_j (j = 1, ..., s) belong to fuzzy values {NB, NM, NS, ZE, PS, PM, PB}. There are r×s possible fuzzy rules to describe the rule I_i→O_j. In Table.1, r = 7, s = 7.

(2) Define maximum and minimum values for each I_i and O_j, thus the upper and lower bounds (width) of each fuzzy variable for possible fuzzy control rule are defined. Then the given data (true values) can be transformed to fuzzy values. Table.1 shows the distribution of fuzzy sets.

Table.1 Distribution of fuzzy sets

O_j $\backslash$ I_i	NB	NM	NS	ZE	PS	PM	PB
NB							
NM							
NS							
ZE							
PS							
PM							
PB							

(3) Establish a no-hidden layer feedforward NN shown in Fig.3. The input and output nodes are NI_i and NO_j, the character N is used to distinguish them from the variables of fuzzy control. The number of input nodes p is the sum of the input and output variables of the fuzzy controller. The number of output nodes q is multiplication of the numbers of input and output fuzzy sets. For example, in Table.1 the input nodes for NN are two (one input and one output) and the output nodes are $7 \times 7 = 49$.
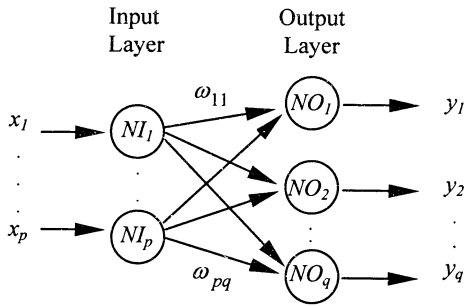
Fig.3 No-hidden layer feedforward NN

(4) By the CL-AVQ algorithm introduced in section 3.3, NN is trained. After that, calculating the number of training data sets that falling into each grid in Table.1, those who have more numbers are the selected fuzzy rules.

This method has been proved that it has the solution and can be converged to it (Jiao [9]). By training the neural network, the clustering algorithm can classify the training data into several well-optimized fuzzy rules.

3.3 CL-AVQ algorithm

The adaptive vector quantization (AVQ), which is used to train NN of Fig.3, divides the input and output spaces into overlapping fuzzy sets. After that, the fuzzy rules are founded through the best relations between input and output fuzzy sets. And competitive learning is described by the differential equation

$$\dot{\omega}_{ij} = S(y_j)(x_i - \omega_{ij}) \quad (2)$$

$\dot{\omega}_{ij}$ is the differentiation of ω_{ij}. And $S(\cdot)$ is a competitive function

$$S(y_j) = \frac{1}{1 + e^{-cy_j}} \quad (3)$$

$$y_j = \sum_i x_i \omega_{ij} \quad (4)$$

x_i is the input to the input layer node. ω_{ij} is the weight that connects between the input node i and the output node j. $S(\cdot)$ likes the step function according to $c \gg 1$. If $S(y_j) = 1$, the output node j wins the competition, otherwise $S(y_j) = 0$. $S(y_j) = 1$ will be satisfied only when the input vector has the minimal distance with the weights to the output node j:

$$\min_j \{ \sum_{i=1}^{p} (\omega_{ij} - x_i)^2 \}^{1/2} \quad (5)$$

where p is the number of the input nodes in Fig.3.

Replacing the differentiation in Eq.(2) by difference, namely, $\Delta\omega_{ij} = \omega_{ij}(k+1) - \omega_{ij}(k)$, the weight between the input node NI_i ($i = 1, ..., p$) and the output node NO_j will be updated by

$$\omega_{ij}(k+1) = \omega_{ij}(k) + S(y_j)(x_i - \omega_{ij}(k)) \quad (6)$$

For those output nodes that fail the competition, the weights will not be modified. Eq.(6) is also written by

$$\omega_{ij}(k+1) = \omega_{ij}(k)(1 - S(y_j)) + x_i S(y_j) \quad (7)$$

Eq.(7) has the following reality: the value $\omega_{ij}(k+1)$ is composed by two parts. One is that the strength of old knowledge $\omega_{ij}(k)$ is declined. The other is the strength of new knowledge x_i is increased. The changes occur within one training data set and both of them are changed exponentially. It can also be considered that $\omega_{ij}(k+1)$ moves to x_i at the exponential speed.

Thus the steps of CL-AVQ algorithm are:
(1) Preparing. Determining the fuzzy feature space and the structure of NN.
(2) Initialization. Initializing weights ω_{ij} by random function.
(3) Competition. Finding the minimal value in Eq.(5) from all the output nodes. The output node j is the winner.
(4) Based on competitive learning, adjusting all the weights to the output node NO_j by Eq.(6).
(5) Repeating steps (3) and (4) until all the data are trained.

4. Experiment

During the experiment, MV is driven to track as closely as possible to the guideline. For comparison, the

same guideline is used to compare the effect with the general fuzzy control presented in Wang [10].

4.1 Experimental object and fuzzification

The structure of MV has been given in section 2. And Fig.4 gives the scene taken by the CCD camera, from which the control parameters can be obtained.

(0, 0)
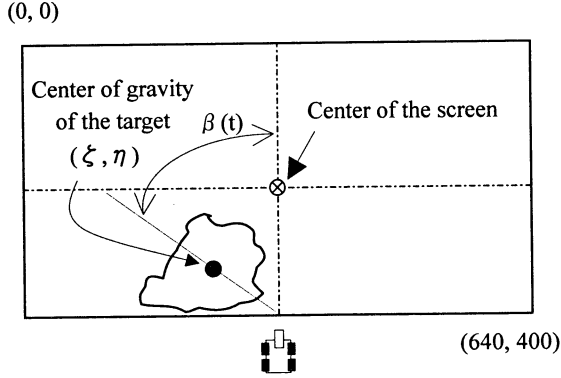
Fig.4   Image processing method

By image processing, the center of gravity of the target (ζ, η) can be obtained. Thus the parameter β (t), which represents the angle between the central line of MV and the target, is converted to the number of pulses and input to the steering motor to steer the front wheels rotating left or right to head to the center (ζ, η).

As introduced in section 2, MV has two degrees of freedom - velocity and angular velocity control. From the driver's point of view, they are mobility and steeribility. This paper gives the experimental result for the steering control. The input variables for the fuzzy control are β (t) and $\Delta \beta$ (t), where $\Delta \beta$ (t) = β (t) - β (t-1). The output of the fuzzy controller is the relative strength ϕ (t) used to control the steering motor.

Assuming that there is little knowledge about the fuzzy rules for the steering control. Based on experience and some presented results, Table.2 gives the assumptive fuzzy rules (candidates), which presents all the possible 25 rules.

Table.2 Fuzzy rules for ϕ (t)

$\Delta\beta(t)$ \ $\beta(t)$	NM	NS	ZE	PS	PM
NM	NM	NM	*NM*	*ZE*	PS
NS	NM	NM	*NS*	*ZE*	PS
ZE	*NM*	*NS*	ZE	*PS*	*PM*
PS	NS	*ZE*	PS	PM	PM
PM	NS	*ZE*	*PM*	PM	PM

β (t), $\Delta \beta$ (t) and ϕ (t) are all divided into five fuzzy values {NM, NS, ZE, PS, PM}. N in fuzzy control means negative, in β (t) means the target is on the left side of the screen, in $\Delta \beta$ (t) means the target is located more left at time t than time t-1, and in ϕ (t) will let the

front wheels turn left. P (positive) is opposite to N.

4.2 Experiment

The sets (β (t$_k$), $\Delta \beta$ (t$_k$), ϕ (t$_k$)) are the real fuzzy control data sampled from experiments based on Table.2, which has 25 candidate fuzzy rules. They are not used as trial and error, but as the inputs to the no-hidden layer feedforward NN shown in Fig.3. The aim of the experiment is to show how to optimize the fuzzy rules based on AFCNN.

The training process is to input these training data sets into the input layer of NN one by one, as introduced in section 3.2. The output layer node j who satisfies Eq.(5) is the winner, and all the weights connect to it will be adjusted by the CL-AVQ algorithm introduced in section 3.3.

To simulate the change of the environment, the guideline is divided into three parts. The first one is a straight line, the second one is a curve and the last past is a horizontal line shown in Fig.5.
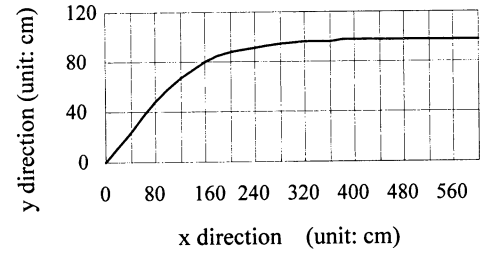
Fig.5   The experimental curve

Compared to the general fuzzy control in Wang [10], where the fuzzy rules and sets were gotten by experience, trial and error, the control result can be given by Eq.(8) and Eq.(9):

$$effect = \frac{\sum\limits_{i=1}^{n} |error|}{n} \qquad (8)$$

$$similarity = \frac{\sum |error_q - error_p|}{\sum |error_q|} \qquad (9)$$

In the above equations, error is the difference between the sampled value and the curve of the guideline, and n is the number of the sampled data. Thus Eq.(8) gives the control result: the more the value effect is small, the more the control is good. In Eq.(9), $error_q$ and $error_p$ represent two kinds of errors controlled by the general fuzzy control and AFCNN, respectively. Thus it gives the similarity of two control methods: the more the value of similarity is small, the more the control by AFCNN is similar to the general fuzzy control.

For the first part of the guideline, $effect_q^{(1)}$ = 2.23, $effect_p^{(1)}$ = 2.11, and $similarity^{(1)}$ = 0.06. Similarly, $effect_q^{(2)}$ = 21.03, $effect_p^{(2)}$ = 11.60, and $similarity^{(2)}$ = 0.36. And $effect_q^{(3)}$ = 3.53, $effect_p^{(3)}$ = 3.36, and

The training data distributed in those 25 grids are given in Table.3. And the italics bold rules in Table.2 is the results after training, which shows that only 13 rules are adopted. The number of data falling into these 13 grids exceeds to a pre-determined value, so they are selected. The strengths of others are so small that they can be neglected (One thing is that based on the fuzzy theory, some control rules are so important that even the numbers of them are small, they should be taken into account).

Table.3 Distribution of training data

Δβ(t) \ β(t)	NM	NS	ZE	PS	PM
NM			.	.	.
NS		.	.	.	.
ZE	.	.	.	.	.
PS		.	.	.	
PM		.	.		

The reason why the fuzzy rules are optimized is that, in Table.2, all the possible control rules are listed, some of them are invalid, and some of them cannot be reached during the experiments (but they may be difficult to be found if only by experience). For example, the top-left fuzzy rule in Table.2 is:

If β (t) is NM and $\Delta \beta$ (t) is NM, then output ϕ (t) is NM (10)

It means that at time t, the center of gravity of the target is on the most left side of MV, and it is more left than time t-1. This can only occur under the following two conditions. One is the control rule at time t-1 was wrong, the other is that the trajectory of the target or guideline varies exceeding the control limit of MV. The first one will not occur if the rules have their rationality. And if assuming that MV can track the trajectory of the target, which is a physically realizable system, the second problem can also be solved. Thus the fuzzy rule of Eq.(10) can be excluded easily.

But before training, or other words, if we have little information about the control object, some fuzzy rules may be difficult to be found their irrationalities (includes the determination of the fuzzy rules, and the center and width of each membership function). They may exist as fuzzy rules in the control system that will never be reached. By AFCNN, they can be analyzed that there is few data belonging to them, and then we can analyze the reason and modify the fuzzy rules to be more reasonable.

By the comparison, when the guideline was a straight line, the control results were almost the same. When the guideline became complicated, control rules by AFCNN were better because they were optimized: the control rules are decreased effectively, so that the control time and program can be improved greatly.

4.3 Analysis

Based on the theory and experiments, AFCNN is proved to be useful in tracking a desired trajectory. Because of its adaptive ability, even if the guideline is a complex curve, as long as it can be tracked, AFCNN can also work well.

Thus the fuzzy rules can be adjusted from the sampled input-output data sets. And if the data are sampled in on-line fashion, fuzzy rules can be modified with the change of real environment in time.

Table.3 gives the distribution of the real fuzzy data sets. From which the following reality is understood:

(1) Initially, MV may have large tracking error to the guideline, so that the fuzzy control rules should be strong. The first several data of control sequences are distributed in the edge grids of Table.3.

(2) Gradually, the fuzzy rules concentrate to the central grids with the decreasing of tracking errors.

(3) The fuzzy rules become complicated when the guideline is a curve than a straight line.

(4) Based on the fuzzy control theory, some control rules are so important that even the numbers of them are small, they should be taken into account (e. g., some edge grids and rules for β (t) = 'ZE').

(5) It also shows that the segmentation of fuzzy space is not average. Edge grids are larger than central grids.

So AFCNN is not only used as the controller to get better control results, but also for fuzzy rule optimizing and fuzzy feature updating.

5. Conclusions

In this paper, the AFCNN system is studied starting from the fuzzy logic and NNs. In fact, one of the main goals of the neuro-fuzzy system is to combine the linguistic representation of fuzzy systems and learning capability of NNs. The AFCNN model uses the learning ability of the neural network and reasoning ability of the fuzzy control for complex, nonlinear and imprecisely defined processes.

By merging the advantages of the neural network, adaptive (clustering) algorithm and the fuzzy control, AFCNN is presented. It can deal with a large amount of training data by the neural network, from these data producing more reasonable fuzzy rules by the adaptive algorithm, and controlling the object by fuzzy control. It is not the simple combination of the three methods, but merging them into one control system.

By AFCNN, both the structural and linguistic complexity could be reduced by refining the rules. Some conclusions are given. First, it is the validity of AFCNN. Selected from the candidates of the fuzzy rules, they are not destroyed but optimized. Second is

its adaptive character. Fuzzy features can be updated with the change of the environment.

Thus AFCNN has the ability to track uncertainly or unknown trajectory of the target or guideline, as long as it is a physically reasonable system.

References

[1] Wasserman P. D. (1989), Neural computing: theory and practice. Van Nostrand Reinhold

[2] Goebel G. (2000), An introduction to fuzzy control systems. http://vectorsite.tripod.com/ttfuzy.html

[3] Abe S. (1995), Neural network and fuzzy: theory and applications. Kindai Scientific Press (in Japanese)

[4] Gray A. (1997), Hybrid systems FAQ. University of Otago at http://divcom.otago.ac.nz:800/COM/INFOSCI/SMRL/people/andrew/publications/faq/hybrid/hybrid.htm

[5] Watanabe K., Izumi K. (1998), Construction of fuzzy behavior-based control systems for a mobile robot. Proc. 3rd International Symposium on Artificial Life and Robotics, Oita, Japan, vol. 2, pp. 518-523

[6] Wang H., Goh C. T. (1999), Fuzzy logic Kalman filter estimation for 2-wheel steer able vehicles. Proc. the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems, Kyongju, Korea, vol. 1, pp. 88-93

[7] Takagi T., Sugeno M. (1985), Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans*. Syst., Man, Cybern., vol. SMC-15, No. 1, pp. 116-132

[8] Ojala T. (1994), Neuro-fuzzy systems in control. Master of Science Thesis. Department of Electrical Engineering, Tampere University of Technology, Finland

[9] Jiao L. (1993), Neural network application and realization. Xi'an Electronic Science and Technology University Press (in Chinese)

[10] Wang X. (1998), Development of intelligent control strategies for a mobile vehicle. Ph.D thesis, Dept. of Electrical and Electronic Engineering, Oita University, Japan

A method for the conversion of the image on convex mirror using artificial life type of function discovery system

Shintaro ADACHI*, Seiichi SERIKAWA**, Kazuki YAMASHITA***
and Teruo SHIMOMURA****

Dept. of Electrical Eng., Kyushu Institute of Technology 1-1, Sensui-cho, Tobata-ku, Kitakyushu, Fukuoka 804 -8550 Japan.

adashin@boss.elcs.kyutech.ac.jp *, serikawa@elcs.kyutech.ac.jp **,
kazu@elcs.kyutech.ac.jp ***, simomura@elcs.kyutech.ac.jp ****

Abstract The convex mirror is one of the important parts for the eyes of robot. However, the shape of the image projected onto the mirror is greatly different from that captured by a usual camera (i.e. plane projection image). For the purpose of image conversion, it is necessary to make the shape on the mirror-surface in high accuracy, but it is not easy. In addition , the camera has to be set up so that the optical axis is corresponding to that of convex mirror. But it takes a lot of time. Thus a method for correcting the distorted image is proposed. This method uses an artificial life type function discovery system. A conversion function, which shows the relationship between the plane image and that of the image on the convex mirror, is automatically calculated by the use of S -System. As the result, the shape of the image can be corrected without distortion, regardles s of the shape of convex mirror and the position.

Key words: artificial life, genetic programming, convex mirror, omnidirectional camera, image distortion

1. Introduction

The omnidirectional camera is used for the eyes of the robot and for the security r ecently. The convex mirror is one of the important parts of the camera. However, the shape of the image projected onto the mirror is greatly different from that projected by a usual camera (i.e. plane projection image). In general, the plane projection image is used for image processing and image recognition, so the image on the mirror has to be converted to the plane projected image. Therefore, the shape of convex mirror is devised so that the image is easily converted into the plane projection image. For the purpose of correct conversion, it is necessary to make the shape on the mirror -surface in high accuracy, but it is not easy. In addition, the camera has to be set up so that the optical axis is corresponding to that of convex mirror. But it takes a lot of time.

In this study, a method for converting the image of convex mirror into plane projection image is proposed. By the use of artificial life type of function discovery system, the plane image without the distortion is obtained regardless of the shape of the convex mirror and the position. J. Koza[1] first proposed the function discovery system based on Genetic Programming (GP). However, the system has some disadvantages. We thus proposed a bug type of artificial life based system to overcome these pr oblems[2][3][4]. We called the system as S -System to indicate the incorporation of the concept of sexual and asexual reproduction in it. In this system, some input data

x and output data *y* are given. The approximate function, which represents the relations hip between them, is automatically found by the process of evolution. We utilize the system. A conversion function, which shows the relationship between the plain image and that projected on the convex mirror, is automatically calculated by the system. As the result, the shape of the image on the mirror can be converted without distortion, regardless of the shape of mirror and the position.

2. Method to correct image distortion

The method for image conversion is mentioned here. In this study, we regard the image conversion as function conversion. Figure 1 shows the relationship between the plain projection image and the image projected on the convex mirror. Let the plain projection image be *A*, and the one on the convex mirror be *B*. We represent the conversion of *A* into *B* as the function *F*, and

Plane image   Image projected
on convex mirror
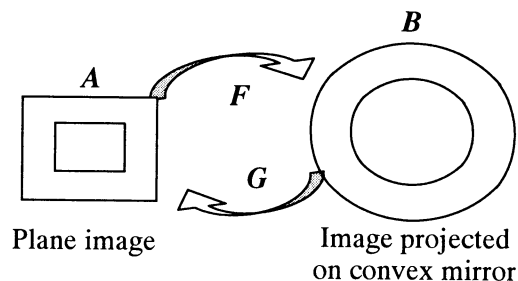
Fig.1 The relationship between the plain projection image *A* and the image projected on the convex mirror *B*.
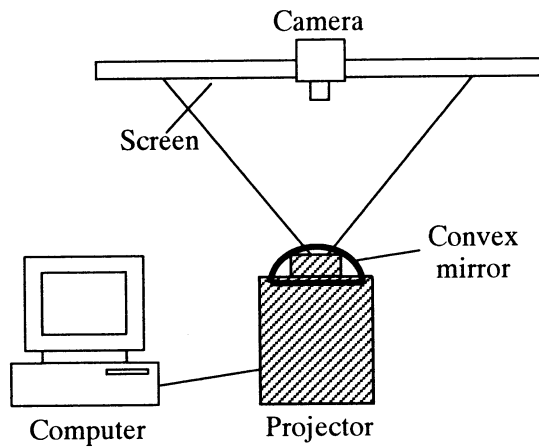
Fig. 2 The structure of setup for finding conversion function *G*.
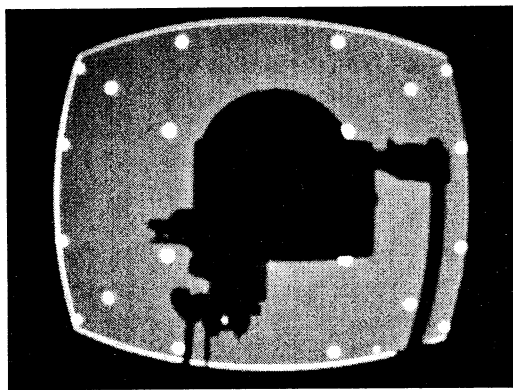
Fig. 3 The image of 20 points projected on the convex mirror.

the conversion of *B* into *A* as the function *G*. If the function *G* is already-known, the plane projection image *A* is obtained by converting *B* by *G*. In this way, if the conversion function *G* is decided, the image on convex mirror can be correctly converted to plane image.

In our investigation, we consider the system as shown in Fig. 2. A personal computer is connected with a projector . On the monitor of the computer, the image without distortion is displayed. The image is displayed onto the screen through the projector. Thus, the image on the screen is regarded as the plane projected image. Since it is projected on the convex mirror, the image is distorted. It is captured by the camera. Therefore, two images such as shown i n Fig. 1 are obtained by this system. That is to say, the image on the screen is corresponding to the image *A*, and the image of camera is *B* in Fig. 1.

3. System flow chart

For finding the conversion function *G*, we adopt the function discovery system calle d "S-system" that uses a bug type of artificial life. In this system, some input data *x* and the output data
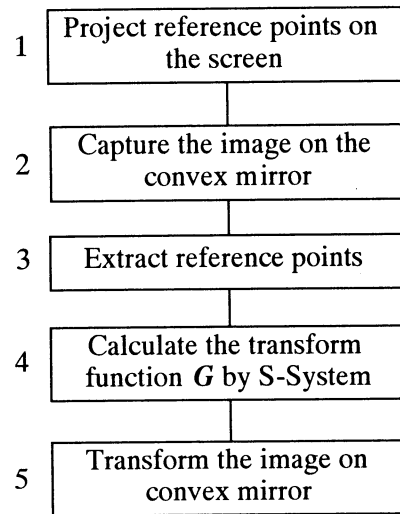
Fig. 4 The flow chart to correct the image distortion.

y are given. The approximate function, which represents the relationship between them, is automatically found by the process of evolution. To find *G*, we display 20 reference points on the monitor of the computer. The points are arranged in equal intervals both vertically and horizontally. The points are displayed onto the screen by projector, and it is projected on the convex mirror. In addition, it is captured by the camera. The image is shown in Fig. 3. We regard the 20 points on the convex mirror as input data *x* and the ones on the screen of the computer as output data *y*. The function *G* that expresses the relationship between *x* and *y* is found by usi ng the S-system.

Figure 4 shows the flow chart to correct the image distortion. It is summarized as follows.

1. A reference image is projected to the screen. In the image, 20 points are arranged in equal intervals both vertically and horizontally. We regard the positions of the points as output data *y*.
2. The image is projected on the convex mirror. The projected image is captured with a digital camera. One of the images is demonstrated in Fig. 3.
3. The 20 points are extracted from the image. We regard the posit ions of the points as input data *x*.
4. The function *G* which represents the relationship between *x* and *y* is found by using the artificial life type of discovery system (S-System).
5. The image is transformed by the obtained function *G*. As the results, the transfo rmed image almost has no distortion.

4. Algorithm of function -discovery

Fig. 5 flowchart

Start
(1) Generation of initial bugs
(2) *Gene* = 0
(3) *Energy*$_p$ = 0 (*p*=1,...,Pop)
(4) Bug *p* = 1
(5) Movement of bug *p*
(6) Capture of data of bug *p*
(7) Find the function ? — Y → End
 — N
(4) *p*=Pop ? — N → *p* = *p*+1
 — Y
(8) *Gene*=*Gene*$_{MAX}$? — N → Descendant generation routine → *Gene*=*Gene*+1
 — Y
End
(9)

Fig. 5   The flowchart of the algorithm of function-discovery using the bug type of artificial life.

Fig. 6 flowchart

SUB
(a) Generation gap Bug No. *Par*
(b) *p* = 0
(c) Selection of bug by tournament strategy
(d) Does the bug sexually reproduce ? — N
 — Y (e) Sexual reproduction
(f) Asexual reproduction
(g) Mutation
(b) *p*=(Pop-Par) ? — N → *p*=*p*+2
 — Y
RETURN

Fig. 6 The flowchart of descendant -generation routine.

The outline of the function -discovery system called S-System is mentioned here.

4.1 Main routine

Figure 5 is the flowchart of the algorithm of function-discovery by the use of a bug type of A-life proposed by us[2]. The flow is summarized as follows.

(1) Numerous bugs with the arbitrary function are generated at random. The number *Pop* is selected from the numerous bugs in order of high fitness.

(2) The generation *Gene* of the bug is set to 0.

(3) The value of the internal energy, *energy*$_p$, of all the bugs is initialized to 0.

(4) The procedures from (5) to (7) are repeated for all the bugs; the bug number ranges from 1 to *Pop*.

(5) The bug *p* moves. This means that the values of constants $\overline{K}$ in the chromosome change slightly. That is to say, the values of $\overline{K}$ are replaced by $\overline{K} + d\overline{K}$, where $d\overline{K}$ is the small change of $\overline{K}$, $\overline{K} = (K_1, K_2, \cdots, K_n)$,

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

(6) The bug *p* catches the observation data (i.e., fitness fit_p of bug *p* is calculated from the observation data).

(7) In the case that fitness fit_p reaches the threshold fitness Fit_{TH}, this algorithm ends. This means a bug has discovered the function *f*.

(8) The algorithm ends when the current generation *Gene* reaches the maximum Generation $Gene_{max}$.

(9) After the descendant -generation-routine is called, *Gene* is added to 1 and the algorithm returns to procedure (3).

4.2 Descendant -generation -routine

The flowchart is displayed in Fig. 6, and is summarized as follows.

(a) Based on the generation-gap, *Par* bugs are selected and they are passed down to the next generation. The elite strategy is adopted for the generation -gap.

(b) The bug number *p* is set to be 0. By the
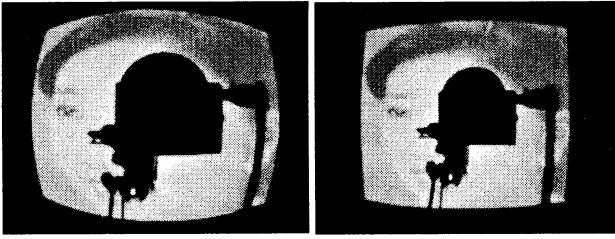
(a) before correction   (b) after correction

Fig. 7 An example of correct of image -distortion
(I).
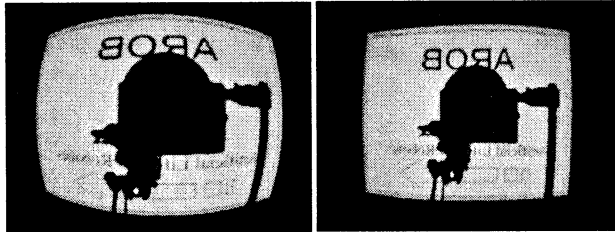
(a) before correction   (b) after correction

Fig. 8 An example of correct of image -distortion
(II).

repetition of the following procedures from (c) to (g), *Pop - Par* bugs are generated.

(c) A bug is selected by the tournament strategy.

(d) The selected bug is judged whether it has the ability to sexually reproduce. In the case that the selected bug has the ability of sexual reproduction, procedure (e) is performed. In the other case, procedure (f) is carried out.

(e) The bug finds its partner, and they produce two children by crossover. Jump to procedure (g).

(f) Two children are produced by asexual reproduction.

(g) A part of the chromosome is changed by mutation at the rate of R_{mut}.

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

In this study, we regard the points on the convex mirror as input data x and the ones on the screen of computer as output data y. The function G that expresses the relationship between x and y is estimated by using the S-system.

5. Application to correct image distortion

To confirm the validity of this method, we projected some figures, and the convert function G is calculated by S -System. By the use of G, the images are corrected. The examples are shown in Figs. 7 and 8. As understood form these images,

the images on the convex mirror are successfully corrected. In this system, the shape of convex mirror and the position are unknown. Nevertheless, the image is corrected by this method. The reason for successful results is that S -System is adopted for finding conversion function G. By the use of S-System, the conversion function G changes dynamically according to the change of the mirror-shape and of the position of camera.

6. Conclusions

A method for correcting the image distorted on the convex mirror is proposed. This method utilizes an artificial life type function discovery system. As the result, the shape of the image on the convex mirror can be corrected to plane projection image, regardless of the shape of convex mirror and the position.

References

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

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

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

(4) T. Shimmomura, K.Yamashita and S.Serikawa "Function discovery system model using non-linear optimization method ", *Proc. Of 6th Symp. On Artificial Life and Robotics (AROB 6th '01)*, pp.321-324 (2001).

Application of Resonance Algorithm for Image Segmentation

Fengzhi Dai [(1)]   Masanori Sugisaka [(2)]
Department of Electrical and Electronic Engineering
Oita University, Oita, Japan
[(1)] daifz@cc.oita-u.ac.jp   [(2)] msugi@cc.oita-u.ac.jp

Abstract

Computer vision and recognition plays more important role in modern intelligent control. In this paper, the resonance theory is proposed for image segmentation. Resonance algorithm is an unsupervised method to generate the region from similar pixels of an image. It tolerates gradual changes of texture to some extent. The purpose of the paper is to propose a practical method for image segmentation, which is always the first step for a real intelligent system. Finally, the experimental results and some considerations are also given.

Keywords: resonance algorithm, image segmentation, gradation in intensity, mobile vehicle

1. Introduction

For an intelligent system (e.g. the autonomous mobile vehicle (MV) experimented in our lab), it is necessary to acquire the information of the external world by sensors, to recognize its position and the surrounding situation. Camera is one of the most important sensors for computer vision.

For image analysis, image segmentation is needed, which means to partition an image into several regions that have homogeneous texture feature. This process is usually realized by the region-based, boundary- based or edge-based method (Castleman K.R. [1]). And from the viewpoint of clustering, it is divided into supervised and unsupervised texture segmentation.

Since before segmentation, for an intelligent control system, we seldom know the feature of the image, e.g. what types and how many types of textures exist in an image, unsupervised segmentation algorithm in needed, although it is more difficult than the supervised method.

We know that human vision can recognize the same texture that has gradient variations of intensity. And many image segmentation methods are proposed by the change of intensity such as in Nakamura T., Ogasawara T. [2] and Deguchi K., Takahashi I. [3], etc.

But these methods may fail to handle wide-ranged gradations in intensity. Also it is usually difficult to give a suitable threshold for pixel-based image processing methods to deal with this gradation. This paper proposes a resonance algorithm, which means that the same texture in an image will be resonated into one region by seed pixels. This method assumes that the feature differences between adjacent pixels of the same texture must be within a tolerable range. Thus the selection of feature distance so as to segment them is important.

In this paper, section 2 gives the resonance theory and algorithm. In section 3, image segmentation and the experimental results are described. And in section 4, conclusions are drawn.

2. Resonance theory and algorithm

2.1 Resonance theory

The resonance theory can be expressed in Fig.1. Assumes that in the scene of space, each point has a mass m. These points are not isolated from each other but inter-connected by inter-force. In Fig.1, two points (a) and (b) are given and the distance between them is d.
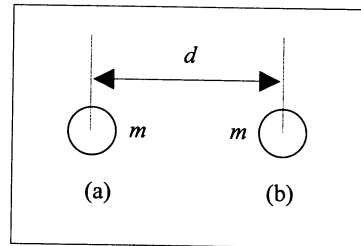
Fig.1   Resonance theory

If an externally sinusoidal force F gives to the point (a), the movement of it can be expressed by

$$m\ddot{y} + b\dot{y} + ky = F\cos(wt) \qquad (1)$$

b and k are system and point parameters, the values of which are not important in the following explanation. The frequency of force F is w. Its stable solution is

$$y(t) = A\sin(wt + \theta) \qquad (2)$$

where

$$A = F / \sqrt{m^2(w^2 - w_0^2)^2 + b^2 w^2} \qquad (3)$$

is the oscillating amplitude. $w_0 = \sqrt{k/m}$ is called the natural frequency.

If the external force has the same frequency as the natural frequency w_0, the amplitude A has the maximal value, and this case is called as resonance between the external force and the point (a).

Since we assumed that the points are not isolated, the motion of point (a) will result in the spreading of its effect to other points that centered on it. This is the resonance among the points, which is the theory for image segmentation. From the analysis by mechanics in He H., Chen Y.Q. [4], the amplitude of point (b) with the distance of d from the point (a) has the feature of

$$A_{distance=d} \propto A / \sqrt{d} \qquad (4)$$

From the above analysis, if we treat a point as the source to resonate and another point can be largely affected by it, the difference between the external frequency and the natural frequency and the distance between two points should be small sufficiently. These two conditions can be satisfied by:

(1) A small threshold is set to assure the difference from the natural frequency is small enough.

(2) The resonance algorithm is utilized within the adjacent points to assure the distance between points is small enough.

2.2 Resonance algorithm

With the spreading of the resonance, points that have the same texture in an image are collected into one region. It seems like the general region growing algorithm introduced in Jähne B. [5], but they are essentially different. Region growing method partitions an image by threshold directly: by defining the maximum and minimum values of each region in an image to segment them. If an image contains complex color (or gray level) gradation, the selection of threshold is difficult.

Differently, the resonance algorithm emphasizes the similarity between the adjacent pixels, not the values themselves. And the resonance can be spread from point to point. Only the sudden changes of feature between adjacent pixels can be regarded as the boundary of different regions.

Define $P_\delta(a, b)$ as the path between the point a and b under the threshold δ (a value to estimate the difference of features between two adjacent points), if there are a sequence of adjacent points connect between a and b, all of which have differences in feature between each other that below δ.

And if s is a point in an image, all points x that satisfies $P_\delta(s, x)$ form the region $R_\delta(s)$. The point s is called the seed. Remember that $R_\delta(s)$ is not centered by the point s, but refers to a region that all points in which have the same features to s, or the difference of features is below δ. Now the principle of resonance algorithm is clear (refer to reference [4]):

From the seed, the adjacent points that belong to the same region under δ are collected until all the points are searched and labeled. From the above definitions and the transfer of resonance, the selection of seed points
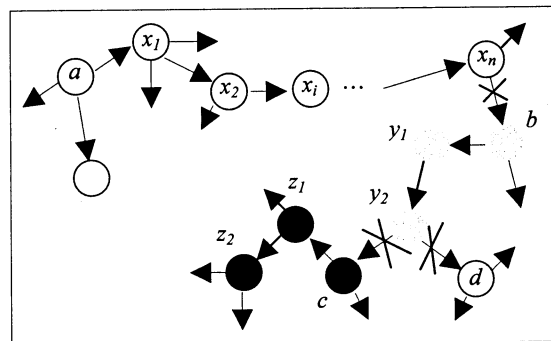
Fig.2   Principle of resonance algorithm

does not influence the segmentation result in an image. Fig.2 gives the expression of the resonant process.

In Fig.2, a is the seed point, from which to resonate the space. By comparing the difference of feature values between it and the adjacent points, all the points that have $P_\delta(a, x_i)$ are labeled to belong to one region, e.g. Regin-1 = $\{a, x_i, i = 1,...,n\}$. Next, from x_n to the point b, the difference between them is larger than δ, b is determined as a point belongs to a new region, e.g. Region-2 = $\{b, y_j, j = 1,...,m\}$, all the points in which are $P_\delta(b, y_j)$. The same is to Region-3 = $\{c, z_k, k = 1,...,l\}$ that are $P_\delta(c, z_k)$ until all the points are labeled.

One thing worthy of mention is the point d. Although it has the same feature as Region-1, since it is far from them, by the resonance theory, it cannot be resonated by any point in Region-1. Thus a new region creates from the point d. That is to say, the number of the segmented regions in an image does not absolutely equal to the number of real different texture regions. In fact, this is not the weakness of the algorithm. Just as this case, we can clearly see that between Region-1 and the point d, there must be some other textures existed.

Thus we may see that from the resonance theory, the resonance algorithm are determined by three important elements: (1) one or several seed points, (2) the features to determine the difference between points and (3) the parameter of threshold δ. The steps of the resonance algorithm are:

(1) Initialization. $R_\delta^0 = \{s\}$.

(2) Segmentation. Find new R_δ^i.

(3) Termination when all the points are labeled.

(4) Result. $R = \bigcup_{i=0}^{M} R_\delta^i$.

The feature of this algorithm is to find a harmonious value within adjacent points, not to estimate the value itself. Fig.3 is the flow chart.

2.3 Selection of elements

Selection of the three elements for the resonance algorithm will be considered in this section. First, we
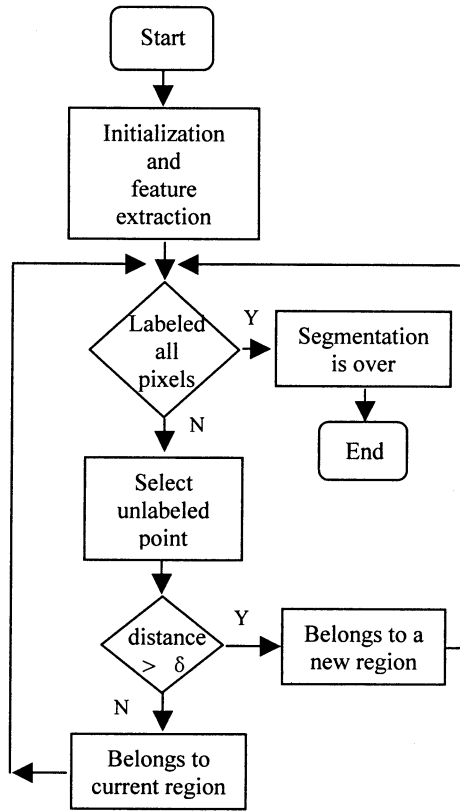
Fig.3 Resonance algorithm

see that from the resonance theory and the transfer of resonance, the initial place of the seed does not influence the segmentation result.

Also, the threshold δ is important in the resonance algorithm. Too large will include surplus regions into one while too small will reject some points that belong to the same region. And different images, or different regions in one image, may have different thresholds. We should propose an automatic selection method for δ :

Since δ is used to partition the different regions in an image, it is rather a range of values than a fixed value - the maximum and minimum value to assure the points have the same features into one region. δ should larger than the *distance* in one region and less than the difference between two different regions.

If δ of different regions in an image are selected well, all the regions are segmented stably. That is to say, when the correct δ varies by a small value, the change of regions in an image is not distinct. Other word is that if the selection of δ is incorrect, a small change of it may vary the area of regions greatly. This is the influence of δ for the region segmentation, and comparatively, it can be used as criterion to estimate δ .

Since the suitable threshold should be greater than the intra-region feature difference and less than the inter-region feature differences expressed in reference [1] and [4], from the initial seeds, the resonance begins from the current region to expend to other regions with the rise of δ .

Fig.4 is an example. In a unit square, there are three different regions: the black, gray and white region. The area of the black square is $S_{black} = \dfrac{1}{2} \times \dfrac{1}{2} = \dfrac{1}{4}$, the area of gray is $S_{gray} = \dfrac{3}{4} \times \dfrac{3}{4} - S_{black} = \dfrac{5}{16}$, and the area of white $S_{white} = 1 - \dfrac{3}{4} \times \dfrac{3}{4} = \dfrac{7}{16}$.

(a)   (b)
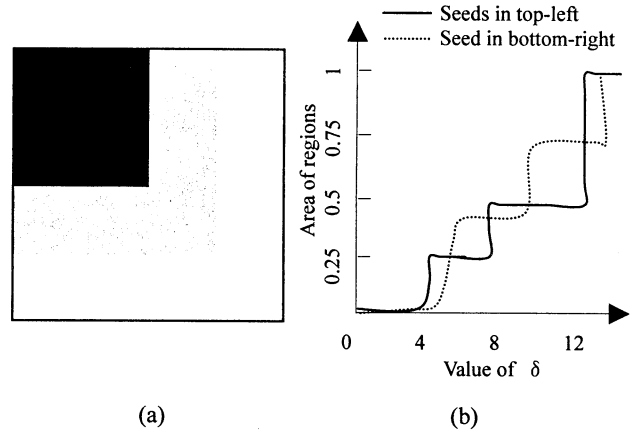
Fig.4 δ -area curve

Fig.4(b) gives the δ -area curves of Fig.4(a) from the seed point selected in the top-left and bottom-right corner respectively.

Another element for resonance is the selection of feature values to estimate whether the *distance* between two adjacent points is below δ or not. In fact, it can be chosen by any features of the image, and the threshold δ is different based on its different selection. In this paper, an 8x8 pixel-based window is adopt so that the average gray level is used as the feature. This feature is illumination-dependent and can be easily compared to other conventional segmentation algorithm.

3. Experiments

3.1 Image segmentation

For multi-object in one image, image segmentation is necessary: locating and isolating the objects from the image and then identifying them. Once isolated, the objects can be measured and classified. The correct segmentation should be that it divides the image S into several independent regions $\{S_1, S_2, ..., S_n\}$, each region represents one kind of textures.

(1) $S = \bigcup\limits_{i=1}^{n} S_i$

(2) $S_i \cap S_j = \Phi$ *for all* $i \neq j$

(3) $P(S_i) = 1$ *for all* i (5)

(4) $P(S_i \cap S_j) = 0$ *for all* $i \neq j$, S_i, S_j *adjacent*

P(x) means the probability of existence of *x*.

Fig.5(a) is always given to compare the different algorithms for image processing. It is the scene of setting the experiment. A strong light is given on the right hand of the black rectangle, and Fig.5(b) is the source image.

(a)   (b)   (c)   (d)
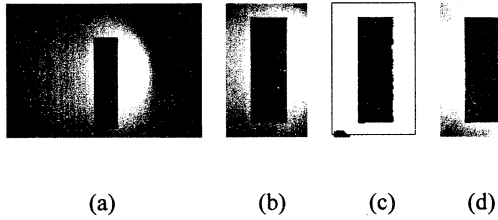
Fig.5 Compared experiment

By the resonance algorithm proposed in the paper, the segmentation result is given in Fig.5(c), where the effect of light is removed, so that the background is in one region. A compared result is given in Fig.5(d) by the general reinforcement histogram algorithm (Agui T., Nakajima M., Kimi J. [6]), from which the background is difficult to be defined into one region.

3.2 Experiment

In this section, compared to the conventional method, the processing for an image taken in real environment will be applied to analyze the resonance algorithm in image segmentation.

Fig.6 shows the source image and the segmentation result. In the original image of Fig.6(a), sky and trees are two main regions, while the color of sky is varied gradually by clouds.

(a)

(b)   (c)
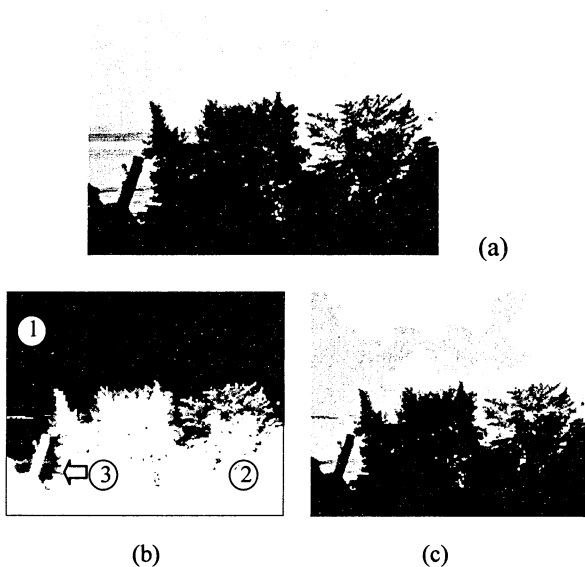
Fig.6 Image segmentation

Fig.6(b) is the result after segmented by the resonance algorithm proposed in the paper. We adopt the average gray level as the feature. And in this experiment, the seed pixel is selected from the top-left corner of the image. The regions are divided into three, for the sky is separated into part ① and ③ by a trunk of tree in the image. Part ② is the region of trees. The effect of clouds is eliminated because the resonance algorithm can deal with gradual changes of intensity.

Also if we do not satisfy that the sky is separated, the part ① and ③ can be combined by the later processing easily, because the features of them are the same. The compared result is given by Fig.6(c), which is produced by the general method of histogram analysis. From it, the effect of clouds cannot be ignored.

4. Conclusions

In this paper, the unsupervised resonance algorithm is proposed for image segmentation, which has the feature to eliminate the effect of gradual changes of texture in intensity to some extent. Before that, the theory and algorithm of resonance and the three elements are considered. The compared result is also given and it shows the advantage of the algorithm: resonance algorithm emphasis the similarity among the adjacent pixels rather than the values themselves.

References

[1] Castleman K. R. (1998), Digital image processing. Original edition published by Prentice Hall, Inc., a Simon & Schuster Company, Press of Tsinghua University, China

[2] Nakamura T., Ogasawara T. (1999), On-line visual learning method for color image segmentation and object tracking. Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems, Kyongju, Korea, vol. 1, pp.222-228

[3] Deguchi K., Takahashi I. (1999), Image-based simultaneous control of robot and target object motion by direct-image-interpretation. Proceedings of The 1999 IEEE/RSJ International Conference of Intelligent Robot and System, Kyongju, Korea, vol. 1, pp.375-380

[4] He H., Chen Y. Q. (2000), Unsupervised texture segmentation using resonance algorithm for natural scenes. Pattern Recognition Letters 21, pp.741-757

[5] Jähne B. (1995), Digital image processing - concepts, algorithms, and scientific applications. The Third Edition. Springer-Verlag

[6] Agui T., Nakajima M., Kimi J. (1991), Image processing with C: PC9801 series MS-DOS. Shokodo Co., Ltd. (in Japanese)

Hough Transform Based Line Segment Detection

Xiongfeng Feng

Zheng Wang

Masanori Sugisaka[1,2], Hidenori Kimura

[1]Department of Electrical and Electronics Engineering, Oita University, Oita 870-1192, Japan (fxf@cc.oita-u.ac.jp)

Shenyang Institute of Automation, Chinese Academy of Sciences, Shenayng, 110016, P.R.China (Email: mousewz@sia.ac.cn)

[2]The Institute of Physical and Chemical Research (RIKEN), Bio-Mimetric Control Research Center, Shimoshidami, Moriyamaku, Nagoya, 463-0003, Japan Department of Complexity Science and Engineering Graduate School of Frontier Science, The University of Tokyo, Tokyo 113-8656, Japan (Email: msugi@cc.oita-u.ac.jp, kimura@crux.t.u-tokyo.ac.jp)

Abstract

Detecting line segments in images is one of the most fundamental problems in image processing. Hough transform is a popular method for line extraction. After detecting the line positions in an image, it is necessary to locate the remarkable line segments on lines. This paper put forward a new line segments detection method by combining Hough transform based approach and line segment grouping approach. Inspired by Newton's Law of Universal Gravitation, an intelligent grouping method for basic line segments is proposed. Effectiveness of the proposed methods is illustrated.

Keywords: Hough transform; Line detection; Line segment detection; Universal Gravitation.

1 Introduction

Line segment is one of the most important primitive objects that are frequently used in robot vision and other image processing fields. There are many papers about line segment detection or line detection in images. Most of the methods proposed in these papers can be divided into two categories – Hough transform[1] based approach and line segment grouping approach [2].

By using Hough transform, it is possible to detect a line where many collinear edge pixels exist, no matter whether they are adjacent or isolated. Therefore, it is a robust line detection method that could be used in noisy images. Moreover, because all the edge pixels along a line are considered, Hough transform has a global nature also. However, original Hough transform does not detect line segments on a detected line. It is necessary to group the

pixels along each detected lines afterwards.

Line segment grouping approach[3]-[5] is based on Elementary Line Segments (ELSs), which are obtained by linking edge pixels and approximating them to piecewise straight lines. These ELSs are used as an input to this approach. Adjacent line segments are grouped according to some grouping criterions and replaced by a new line segment. This process is repeated until no new line segment occurs[2]. The advantage of this approach is it gives out line segments as the result, which is not directly available in the Hough transform based approach. However, the process from obtaining ELSs to grouping them into line segment is quite local and cannot guarantee a global optimal result.

In this paper, an integrated approach, which combines the aforesaid approaches' advantages, is proposed. Firstly, Hough transform is used to detect the most obvious lines globally. Secondly, continuous edge pixels along the detected lines are linked to produce Basic Line Segments (BLSs). Thirdly, BLSs are grouped according to a set of intelligent grouping criterions, which is inspired by Newton's Law of Universal Gravitation, to produce line segments of global nature.

2 Hough transform and radon transform for line detection

The basic idea of Hough transform could be explained by following.

The duality between points and lines could be observed in Fig. 1, where points P_1 and P_2 in x-y coordinate system correspond to lines L_1 and L_2 in k-b coordinate system, and points P_0 in k-b coordinate system correspond to lines L_0 in x-y coordinate system. When L_0 is vertical to x-axis, k_0 will

be infinity and cause trouble on calculation. Therefore, it is very common to use point-sinusoid dualistic transformation instead of point-line dualistic transformation as following:

$$P=x*\cos(A)+y*\sin(A) \qquad (1)$$

Formula (1) is the common description of Hough transform in literatures. In A-P coordinate system, formula (1) gives a sinusoid. It is easy to prove, similar to duality in Fig.1, that all the points on a line in x-y coordinate system correspond to a point in A-P coordinate system.

When using Hough transform to detect line in an image, generally, the first step is to change the image into an edge image. There are many edge detection methods available in literature, such as Sobel method, Prewitt method, Roberts method, Canny method and Laplacian of Gaussian method, etc.[6] The second step is to discretize the parameter space in A-P coordinate system for a certain range of A and P. For every pixel in edge image, pixel coordinate (x,y) is used to calculate the values of P using formula (1), when A varies from 0~180° with a certain step. To every calculated point of (A,P), a corresponding accumulator (every pair of A and P in discretized parameter space corresponds to an accumulator) will add one to itself. After all edge pixels are processed, checking the highest values in the accumulators will give the position information of the most obvious lines.

In Image Processing Toolbox of Matlab, Hough transform is implemented by Radon transform[6]. Supposing f(x,y) describes an edge image in x-y coordinate system, in general, the Radon transform of f(x,y) is the line integral of f(x,y) parallel to the y axis, as following:

$$R_\theta(x') = \int_{-\infty}^{\infty} f(x'\cos\theta - y'\sin\theta, x'\sin\theta + y'\cos\theta)dy' \qquad (2)$$

where $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$

Fig.2 gives the geometry of the Radon transform. Given a value of θ, by using formula (2) in discrete way, values of (R,x') could be calculated. After calculating matrixes R and x' with θ varying from 0~180° with a certain step, by sorting matrix R and referring to corresponding value of matrix x', position information of the most obvious lines will be available.

3   Line segment detection

Although the line detection by using Hough transform has the advantage of global nature, it does not give exact information of line segment detection. The following sessions describe how to locate line segments on the detected lines.

3.1 2-D filter for detection of BLSs

In our method, similar to the line segments grouping approach, we also need to group some element line segments to form longer and meaningful line segment. However, instead of obtaining ELSs by linking edge pixels locally, the element line segment used here is obtained by searching and linking of the pixels on or near the already detected line (current line). We name this kind of element line segment as Basic Line Segments (BLSs).

The searching process of BLSs is along the current line. Due to computational error or image noise, the searching may not necessarily capture edge pixels that actually belong to the current line. To make up for this weakness, 2-D filter is introduced to filter along the current line. After filtering and appropriate thresholding, new edge pixels that will help to detect line segments will appear on the current line. These new appeared pixels present the influence of the edge pixels that are nearly collinear with the current line.

After that, it is straightforward to obtain the BLSs by counting adjacent edge pixels along the current line. Every set of adjacent edge pixels will form a BLS. To facilitate succedent work, the beginning and ending position as well as the length information of every BLS are recorded.

3.2 Grouping of BLSs

Now we have all the detailed information about BLSs on the current line. Some of the BLSs should be grouped to form meaningful line segments. To this kind of clustering problem, human being has straightforward feeling about which should be grouped and which should not. To do it by programming, however, is not an easy thing. Intuitively, when people are thinking about whether two line segments should be linked to form a new line segment, usually two factors will cause attention - the length of both segments, and the distance between them. The longer the distance, the shorter the segments, the lower the possibility to link them. This analysis gives us an inspiration to use Newton's Law of Universal Gravitation for reference. The Law of Universal Gravitation is as following,

$$F=c*M_1*M_2/R^2 \qquad (3)$$

where c is a constant, M_1 and M_2 are mass of two objects, R is the distance between them, and F is the gravitational attraction.

Based on the above analysis, analogously, we give a

formula for calculating the "attraction" between two line segments as following,

$$F_1 = c_1 * L_1 * L_2 / R_1^2 \qquad (4)$$

where c_1 is some constant, L_1 and L_2 are length of two adjacent line segments, R_1 is the distance between them, and F_1 means the "attraction" between two segments. Obviously, if we take F_1 as a measurement of the possibility of linking the adjacent line segments, the definition of formula (4) will meet the common thought of human being.

Since the detail information of each BLS is already known, the grouping process will always begin from the longest line segment (current BLS). Therefore, formula (4) could be simplified to

$$F_1 = c_1 * L_2 / R_1^2 \qquad (5)$$

Other consideration from common knowledge in grouping line segments include,

a) Comparing distance to its adjacent BLS's length, if the distance is large, the possibility of linking this BLS should be small.

b) Even if the distance is relatively small when comparing to its adjacent BLS's length, if the absolute length of distance is large, the possibility of linking this BLS still should be small.

Formula (5) meets these two considerations well.

Till now, considering we are trying to group many BLSs in one direction only (for ease of explanation), some grouping criterions could be made by using formula (5) as following,

a) When $F_1 < \alpha$, the current BLS should absorb the adjacent BLS to form a new BLS, where α is an empirical constant.

b) When $F_1 > \beta$, the current BLS should be considered as an already grouped line segment, where β is another empirical constant.

c) When $\alpha < F_1 < \beta$, the current BLS should try to link its second adjacent BLS, and the distance used in the calculation should be the distance adjacent to the BLS to be linked.

The reason of criterion c) is that the isolated BLSs between two BLSs to be linked, could be treated as bridges of linking and distances between these isolated BLSs and current BLS could be omitted temporarily.

Generally, current BLS needs to link other BLSs on both sides of itself. It is easy to modify the above criterions for two-side grouping. After the current BLS become a line segment, it is necessary to check how much percent of the edge pixels do the detected line segments have on the current line. If the percentage is above some threshold (85%, for example), the grouping process on the current line should be terminated. This means we only interested in detecting obvious line segments.

4 Illustration of proposed method

Effectiveness of the proposed methods could be shown by Fig.3-5. Fig.3 is an image taken from a corridor by video camera on a mobile robot. Fig.4 is the corresponding edge image of Fig.3. By using the methods presented above, obvious line segments are detected and shown in Fig. 5. Note some line segments are from same line.

5 Conclusion

Hough transform based approach and line segment grouping approach are introduced. A new line segment detection method by combining these two approaches is put forward. Inspired by Newton's Law of Universal Gravitation, an intelligent grouping method for basic line segments (BLSs) is proposed. Effectiveness of the proposed methods is illustrated.

References

[1] R. Wang, Image understanding, Publishing Company of Univ. of National Defence Science and Technology, 1994

[2] Jeong-Hun Jang and Ki-Sang Hong, "A New Line Segment Grouping Method for Finding Globally Optimal Line Segments", International Workshop on Biologically Motivated Computer Vision (BMCV 2000), pp. 397-406, Seoul, Korea, May 15-17, 2000.

[3] M. Boldt, R. Weiss, and E. Riseman, "Token-Based Extraction of Straight Lines", IEEE Transactions on System, Man, and Cybernetics, Vol.19, No.6, pp. 1581-1594,1989.

[4] P. F. M. Nacken, "A Metric for Line Segments", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.15, No.12, pp.1312-1318, 1993.

[5] E. Trucco and A. Verri, "Introductory Techniques for 3-D Computer Vision", Prentice Hall, pp.114-117, 1998.

[6] The Mathworks Inc., Image Processing Toolbox User's Guide (Version 2), 1999
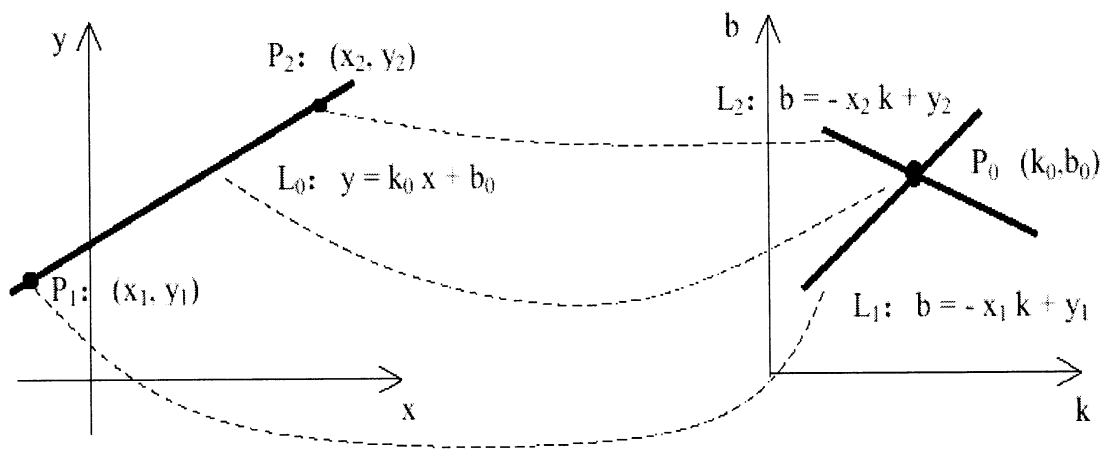
Fig.1 Point-line dualistic transformation
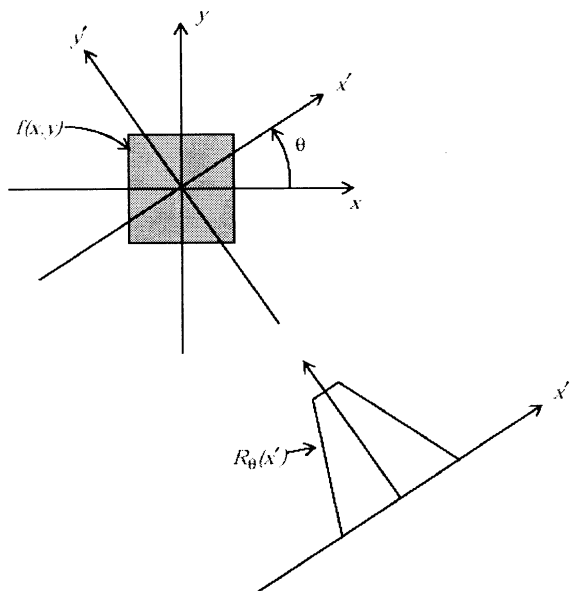
Fig. 2 Geometry of the Radon transform
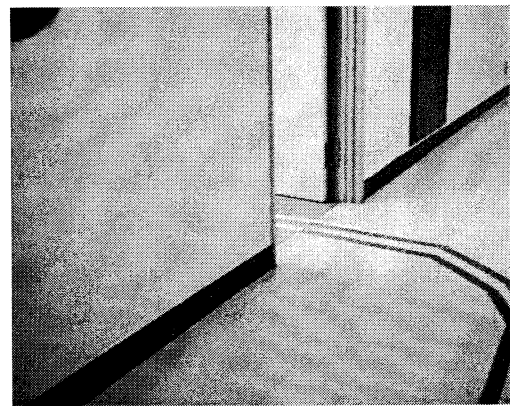
Fig.3 Original image
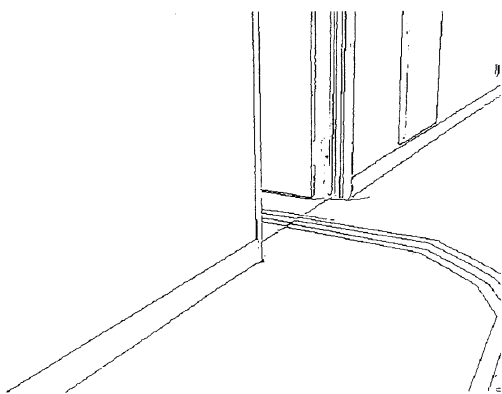
Fig.4 Edge image
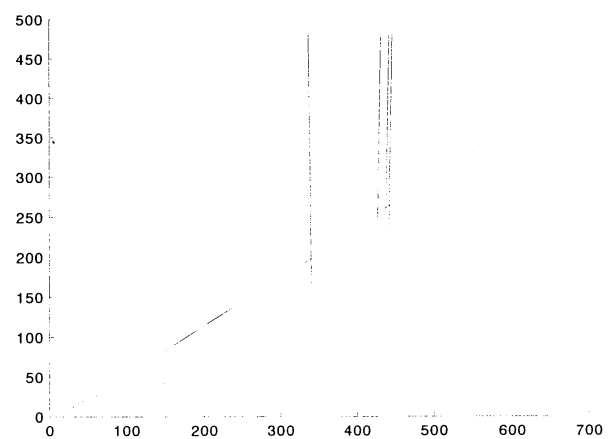
Fig. 5 Detected line segments

Universal constructor to build a Tierran machine structure

Shuichi Matsuzaki†‡, Hideaki Suzuki‡, Minetada Osano†

†Graduate School of Computer Science and Engineering, Aizu University
Aizu-Wakamatsu City, Fukushima 965-8580 Japan

‡ATR Human Information Science Labs
2-2-2 Hikaridai Seika-cho, Soraku-gun, Kyoto 619-0288 Japan

Abstract

In constructing an Alife system with a high potential for evolution, one of the points that we should design most carefully is the machine structure, that is, the architecture of the system controlling creature (CPU) operations. This paper proposes an approach to modify a machine structure of Tierra in order to get higher evolvability through 'coevolution'. The machine structure we propose in this paper is stimulated by the self-reproducing model proposed by von Neumann, which enables self-reproduction in a simple architecture. Our machine structure is compared and discussed with the original Tierra and Neumann's model in the last part of the paper.

1 Introduction

In the development of an Artificial Life (Alife) system, it is important that the system be designed so that 'emergence' (appearance of higher functions among elements) might happen. It is necessary for us to prepare a good artificial environment during system construction because the possibility of emergence happening in the system usually depends on the quality of it prepared by the designer. The actual biological system provides a good example for deciding what to prepare in an Alife system. Today, it is widely accepted that because the ancient earth happened to be placed into an appropriate state, life emerged and a variety of complex life forms evolved. If we could identify the conditions necessary for this evolution [4], it would greatly help the design of a 'good' Alife system. Artificial environments prepared for an Alife system in a computer usually have a two-layered structure: the information space (hardware layer), which cannot be changed during evolution, and the reaction rule set (software layer) which can be changed during evolution. evolvability of the system. Alife is an approach that uses a policy of "minimum implementation" in the design. In order to have a system with high evolvability, we should prerpare the hardware layer so that the software layer might be changed/optimized by evolution as much as possible. From this point of view, we propose optimizing the instruction set (the software environment) of Tierra designed by Ray [1]. Though in the original experiment of Tierra [1, 2], the instruction set was pre-programmed and unchanged during a run, if we were able to equip creatures (CPUs) with different instruction sets and make those creatures compete with each other, then we might be able to optimize the instructions through evolution. This paper reports the results of the first step of this approach. To modify and optimize the instruction set programmed in the system, our research aims to redesign the machine structure of Tierra in order to make the instruction set changeable. The instruction set is fetched from the code data subject to mutational modifications. As a result, the machine structure is greatly changed compared with the original Tierra. The architecture of Tierra was reconstructed based on the idea of the self-reproducing machine proposed by von Neumann in the 1950's.

In the following section, we first describe the machine structure of Tierra and its problems by comparing it with von Neumann's machine structure. Section 3 gives the structure of the redesigned Tierra and shows details of each part of the structure. In section 5, we present our concluding remarks and future directions.
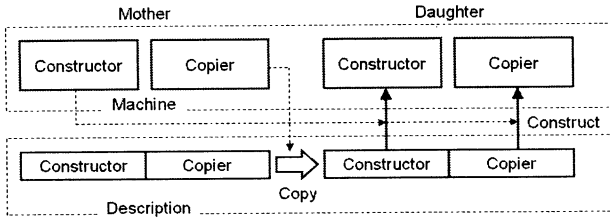
Figure 1: John von Neumann's Self-reproducing machine structure

Figure 2: Machine structure of Tierra

2 Self-reproducing machine structure

Since Ray proposed Tierra in 1992, it has been recognized as the most typical and representative Alife model. The Tierra simulation experiment produced stimulating results in the behaviors of virtual creatures. However, we believe that Tierra is not yet completed, at least from the viewpoint of self-reproduction.

In connection with this matter, here we introduce another Alife model following the excellent suggestions proposed by von Neumann (Fig. 1). He designed a self-reproducing machine comprised of a 'machine' with the function of a universal constructor and copier, and a 'tape' consisting of descriptions of both the constructor and copier (conceptually, these objects consist of descriptions that might be called a 'description' in the following). The constructor manufactures another machine called a 'replicant', and the copier makes a copy of the descriptions for the replicant. Both the constructor and copier use the data of the description in each working process. Although von Neumann introduced the tape as just a storage device never changed, we consider this self-reproducing machine can be regarded as a model of evolution if we suppose that the tape is an "artificial genome" and is changeable in self-reproduction. Additionally, there is a possibility that this machine can be a powerful Alife model because its two kinds of description, that is, a constructor and copier, can evolve concurrently. competing with each other.

Here, we return to the discussion of Tierra. Figure 2 shows the machine structure of Tierra. In Tierra, the tape in von Neumann's model can be regarded as a core memory and the copier as a mixture of the functions of a program, a CPU, and an operating system (OS). Although the role of the constructor is achieved by the functions of the CPU and OS, these structures are implicitly programmed in the system. Therefore, a creature in the Tierra create its replicant's CPU by simply executing one instruction ('divide'), but almost all of the procedures necessary for the construction are executed by their OS. Note that in the simulation, the programs embedded in the OS cannot be changed by any operations. This indicates the impossibility of optimizing the isntruction set in the current machine structure.
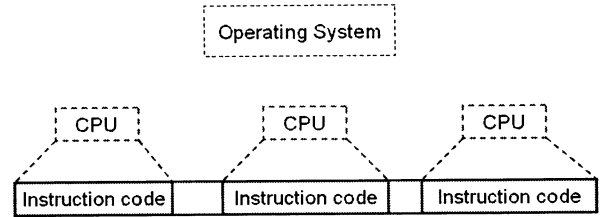
Noteworthy features of the Tierra structure are listed as follows.

- The instruction code serves mostly the same role as the function of the copier, but there is no explicit constructor (it is programmed in the OS).

- In the simulation of Tierra, only the instruction code evolves, and the other functions are embedded in the operating system.

3 Machine structure of Tierra redesigned

The machine structure we propose is shown in Fig 3. This structure is redesigned based on von Neumann's machine structure and has some structures revised from the original machine structure of Tierra. Although this architecuture is not so complicated, some functions must be explained because they do not normally appear in Tierra.

As we discussed, since coevolution between the instruction code and instruction set is our primary objective, a description of the instruction set is fetched from the core to make it changeable.

The structure is divided into two parts: 'Machine' and 'Description'. The machine has the structure composed of a single 'control memory' and other 'registers' (see Table 1). The description has the codes of 'microcodes'(Fig 4), 'constructor'(Fig 5), and 'copier'.
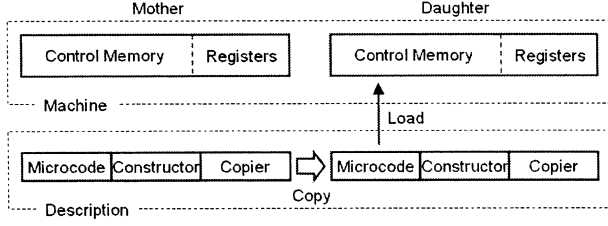
Figure 3: Machine structure of Tierra redesigned

The microcode is a description of a set of microoperations, which consists of the coded thirty-two instruction set. The copier, a program for the copying the description, is almost equal to the ancestor program of original Tierra, except for the function of the constructor. The constructor is a sequence of instructions which executes the 'load' instruction (see Table 5), which loads the microcodes into the control memory. Then, using the data in the control memory, the machine can recognize the code written in the constructor/copier and execute the instructions. Finally, the noteworthy topics in this reconstruction are listed as follows.

- The machine and the description are explicitly defined in the structure.

- Although the machine is not divided into two parts like von Neumann's machine structure, the machine operates as both constructor and copier.

- The microcodes, namely the instruction set, are explicitly described in the description. That makes the instruction set changeable.

4 Discussion

This paper proposed a machine structure of Tierra that enables coevolution between the program and the instruction set. We have already finished the basic design of the proposed architecture of the revised Tierra (Table 1, Fig 4, and Fig 5); and in the next step, we are planning implementing the proposed system and conducting computer simulation in order to examine the validity and evolutionary ability of the proposed design.

```
/* Microoperations of the Tierran instruction set */
Fetch    (CMA==0): CMA ← M
  nop0   (CMA==1): CMA ← 0,  MA ← MA + 1
  nop1   (CMA==2): CMA ← 0,  MA ← MA + 1
pushax   (CMA==3): ST  ← A
         (CMA==4): SP  ← SP + 1, CMA ← 0, MA ← MA + 1
         ...

  movii  (CMA==21): ST  ← MA
         (CMA==22): MA  ← B,  SP ← SP + 1
         (CMA==23): ST  ← M
         (CMA==24): MA  ← A
         (CMA==25): M   ← ST
         (CMA==26): SP  ← SP - 1
         (CMA==27): MA  ← ST
         (CMA==28): CMA ← 0, MA ← MA + 1
 sub_ab  (CMA==29): C   ← A - B, CMA ← 0, MA ← MA + 1
 sub_ac  (CMA==30): A   ← A - C, CMA ← 0, MA ← MA + 1
  inc_a  (CMA==31): A   ← A + 1, CMA ← 0, MA ← MA + 1
         ...

  ifz    (CMA==38)&&(C!=0): MA  ← MA + 1
         (CMA==39): CMA ← 0,  MA ← MA + 1
  jmp    (CMA==40): MA  ← MA + 1
         (CMA==41)&&[(M==1)||(M==2)]: PST ← M
         (CMA==42)&&[(M==1)||(M==2)]&&(PST==1): PST ← 2
         (CMA==43)&&[(M==1)||(M==2)]&&(PST==2): PST ← 1
         (CMA==44)&&[(M==1)||(M==2)]: PSP ← PSP + 1, CMA=40
         (CMA==45): S ← PSP
         (CMA==46)&&(S!=0): PSP ← 0, SF ← 0
         (CMA==47)&&(S!=0)&&(SF==0) : MA ← MA + 1
         (CMA==48)&&(S!=0)&&(SF==1) : MA ← MA - 1
         (CMA==49)&&(S!=0)&&(PST==M)&&(PSP<S): PSP ← PSP + 1,
                                     MA ← MA + 1, CMA=49
         (CMA==50)&&(S!=0)&&(PST==M)&&(PSP==S) : S ← 0, PSP ← 0
         (CMA==51)&&(S!=0) : PSP ← 0,  ST ← MA
         (CMA==52)&&(S!=0)&&(SF==0) : SF=1, SP ← SP + 1, CMA=48
         (CMA==53)&&(S!=0)&&(SF==1) : SF=0, SP ← SP - 1, CMA=47
         (CMA==54): PST ← 0, CMA ← 0, MA ← MA + 1
```

Figure 4: Microcodes for several example instructions. There are 157 micooperations for translating thirty-two instruction sets. Once the instruction written in the constructor or copier is interpreted, the corresponding microoperation is executed in sequence.

```
CM      /* control memory */
CMA     /* address pointer of CM */
A       /* address register */
B       /* address register */
C       /* numerical register */
D       /* numerical register */
E       /* top address */
F       /* end address */
S       /* template size */
SF      /* search flag (0/1) */
ST      /* stack */
SP      /* stack pointer of ST */
PST     /* stack for template */
PSP     /* stack pointer of PST */
M       /* core memory */
MF      /* status of M (0/1) */
MA      /* address pointer of M */
```

Table 1: List of the registers/stacks and their primary functions. There are many registers added from the original register structure of Tierra.

```
/* Original ancestor' s code */

zero; put zero in cx
nop0 ; beginning marker
nop0 ; beginning marker
nop1 ; beginning marker
nop0 ; beginning marker
load; copy micro-operation into Control Memory
|ifz; if cx ==0 perform next instruction, otherwise skip it
jmp; bidirectional jump to template below
nop1 ; beginning marker
nop1 ; beginning marker
nop0 ; beginning marker
nop1 ; beginning marker
```

Figure 5: Description of the constructor. This is the program to load the replicant's microcode into the control memory.

References

[1] Ray, T. S. " An approach to the synthesis of life" In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, Eds., Artificial Life II. Redwood City, CA: Addison-Wesley (1992) pp.371-408

[2] Ray, T.S., Hart, J.: Evolution of differentiated multi-threaded digital organisms. In: Adami, C., Belew, R.K., Kitano, H., and Taylor, C.E. (eds.): Artificial Life VI: Proceedings of the Sixth International Conference on Artificial Life, MIT Press, Cambridge, MA (1998) pp.295-304

[3] Suzuki, H. "String rewriting grammar optimized using an evolvability measure" In: Kelemen, J., Sosik, P. (eds.): Advances in Artificial Life (6th European Conference on Artificial Life Proceedings), Springer-Verlag, Berlin (2001) pp.458-468

[4] Suzuki, H., Ono, N.: Several information space conditions for the evolution of complex forms of life. In: Sugisaka, M., Tanaka, H. (eds.): Proceedings of the Seventh International Symposium on Artificial Life and Robotics (AROB 7th '02) Vol. 1 (2002) pp.175-178

[5] von Neumann, J.: Theory of self-reproducing automata. University of Illinois Press, Urbana. Edited and completed by A.W. Burks (1966)

[6] Moshe Sipper.: Fifty Years of Research on Self-Replication: An Overview. Artificial Life 4(3): 237-257 (1998)

[7] Matsuzaki, S., Suzuki, H.: Tierra instructions implemented using string rewriting rules: Proceedings of the Fifth International Conference on Human and Computer (HC-2002) pp.167-172

[8] Suzuki, H., Ono, N.: Universal Replication in a String Rewriting System: Proceedings of the Fifth International Conference on Human and Computer (HC-2002) pp.179-184

Evolution from molecules to proto-cells in an inhomogeneous environment.

Naoaki Ono

ATR-HIS, 2-2-2 Hikaridai, "Keihanna Science City"
Kyoto, 619-0288, Japan

Abstract

This paper presents a study to elucidate the evolution of primitive metabolic systems using a model of artificial chemistry that simulates the reactions and diffusion of abstract molecules. It is supposed that pre-cellular chemicals were able to survive only in the limited regions where the environment was suitable for sustaining their self-replication. Therefore, in this study, we focused on the evolution of the primitive metabolic system under spatially heterogeneous conditions in which the supply of resources is high enough in a small region, and low in other regions. The result shows that the catalysts that produce membrane molecules can evolve under this limited condition, proto-cell structures emerge automatically, and through cellular selection, the new catalysts spread and finally become dominant. In fact, this condition induced evolution rather than hindered it. Namely, the evolution of new catalysts and the organization of proto-cells took place at the marginal regions where it became difficult to survive.

Keywords: self-organization, self-replication, artificial chemistry.

1 Introduction

In order to understand the origin of life, various theoretical models have been proposed to analyze the evolution of self-reproducing molecules. Because there is little evidence to show how molecules appeared in the earliest metabolic systems, artificial chemistry, that is, a model that simulates physiochemical interactions among abstracted chemicals, is a useful tool for understanding what could have happened in the supposed environment of the primordial earth. For example, Computational Autopoiesis is a model of a self-maintaining cell based on a cellular automaton [2, 1]. It is worth noting that the creation of membranes, that is, the boundary that distinguishes a cell from its surroundings, is an essential feature of living organisms [5]. The organization of proto-cells in an abstract space was shown in an artificial chemistry embedded in a planer graph [4].

Lattice Artificial Chemistry (LAC) is a model proposed by Ono and Ikegami [3]. Using this model, they demonstrated that self-maintaining and self-reproducing proto-cells emerge spontaneously through pre-cellular chemical evolution. To investigate more detailed conditions under which the evolution of proto-cells is possible, we studied this evolution under various conditions using extended LAC. We are interested in not only the range of the parameters but the robustness against the spatial differences or time change in the environment.

It has been supposed that the resources of the earliest self-replicating molecules were supplied from an abiotic source, e.g., lightning, ultraviolet light radiation, or hydrothermal deposits in the sea bottom. However, these supplies on the primordial earth were probably rather unstable or limited to some specific regions spatially. For example, no energy is supplied from the sun at night, and the temperature and the concentration of molecules will fall fast as it becomes distant from the source deposit. Such insufficient conditions should be taken into account, to understand the evolution of primitive metabolic systems. In this report, we present the study of an inhomogeneous environment where the resources are supplied locally. The dependency over time change in the environment will be discussed elsewhere.

2 Model

LAC is based on a discrete and stochastic dynamics that simulates the chemical reactions and spatial interactions of abstract chemicals under non-equilibrium conditions. Chemicals are represented by abstract particles placed on a lattice of reactions sites. The reactions and diffusion of chemicals are expressed by the probabilistic change of the particle type and random walks, respectively. These transition probabilities are given as a function of the potential of the particles.

We introduced 10 species of autocatalysts $(A_0 \ldots A_9)$. An autocatalyst A_i catalyzes the reproduction of another autocatalyst A_j consuming a resource particle (X_A). When an A_j is reproduced, there is a small probability that a variant $A_{j\pm1}$ may be produced. All autocatalysts can catalyze the reproduction of each other equally and share a common reproduction rate, but the activities of the production of membranes differ with species. Membrane particles (M) are produced from another resource particle X_M and they organize stretched clusters according to the interaction with water particles (W) spontaneously. Most of the autocatalysts do not produce M, or produce just a little, but a few species can produce M more efficiently. The activity of membrane production compared to that of the most effective one (A_9) is shown in Table 1.

Table 1: The activity of membrane production.

sp.	$A_{0\ldots3}$	$A_{4\ldots6}$	$A_{7,8}$	A_9
rate	0	0.01	0.1	1

These particles, except W, automatically decay into waste particles $(Y_A$ and $Y_M)$. However, there is a source of resource particles to keep the system in a non-equilibrium. In this study, we assume that there is a gradient of resource supply as illustrated in Fig. 1. The supply is enough to sustain the metabolism near the top end, but it becomes lower toward the bottom.
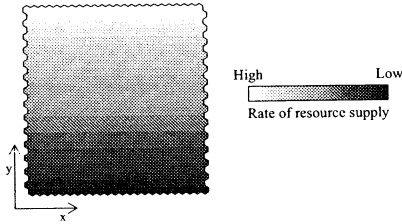
Figure 1: The rate of resource supply becomes the highest $(S_X = 8.0)$ at the top border and becomes the lowest $(S_X = 4.0)$ at the bottom border. The left and right boundaries are periodic.

3   Result

Snapshots from a simulation are shown in Fig. 2. Figure 2(1) shows the initial state. At the top where the resources are enough, there is a cluster of catalysts A_0. The remaining region is filled with water, resource, and waste particles. At first, the autocatalysts spread reproducing themselves, but they cannot reach the lower half where the resources are not enough (Fig. 2(2,3)). As the autocatalysts A_0 reproduce, other species appear through mutations. As species with higher activity begin to produce membrane particles, fragments of membranes begin to appear (Fig. 2(4)).

Next, the new evolution starts at the frontier where the catalysts disappear. Because the rates of reactions depend on the density of the catalysts, it is necessary for their survival to keep their density high. However, at the frontier, the catalysts are always lost by diffusing into regions where they cannot sustain the metabolism. If there is a fragment of the membrane near the frontier, it somehow obstructs their diffusion so that they can metabolize faster. Thus, there sometimes arises a growing region covered by a fragment of the membrane. Note that this region probably contains more catalysts with a higher activity of membrane production. This implies that the catalysts that produce membrane particles may have a little advantage over the others in these regions.

If the region with the membrane grow, the possibility that some part of the region is wholly enclosed by a membrane also increases. Once an enclosure is organized, it begins to maintain itself by producing new molecules inside it (Fig. 2(5)). When a proto-cell grows, it assimilates resources from the environment and accumulates within it by metabolizing them. It divides itself into daughter cells automatically when it grows large enough, and the daughter cells grow and reproduce themselves recursively. Due to these features of self-maintenance and self-reproduction, we call this a "proto-cell" hereafter (Fig. 2(6,7)).

Because the membrane can keep the density of the catalysts within it high, proto-cells can survive in the regions where the metabolism was impossible at first (Fig. 2(8,9)). On the other hand, since the resources are taken by the proto-cells, the catalysts outside the cells gradually become extinct. At last, the whole region is filled with the proto-cells (Fig. 2(10-12)).

Figure 3 shows the time series of the populations of the 10 autocatalysts. At first, there are only A_0, but the diversity increases soon through mutations. Before the organization of the proto-cells, the populations tend to converge into the same size. However, as proto-cells appear and spread, the population of A_9, which produces M most efficiently, increase and become dominant at last.
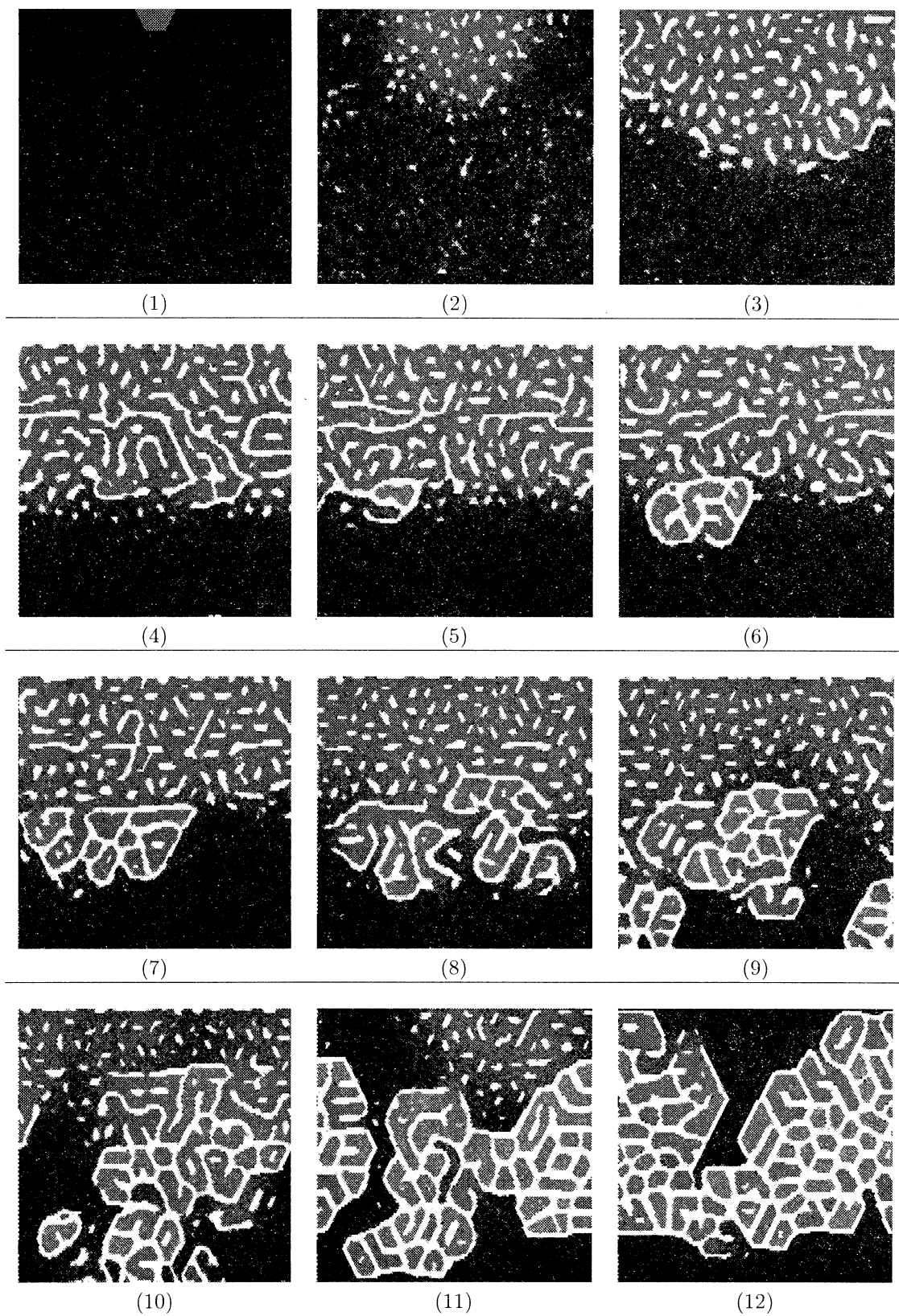
Figure 2: The spontaneous emergence and evolution of proto-cells
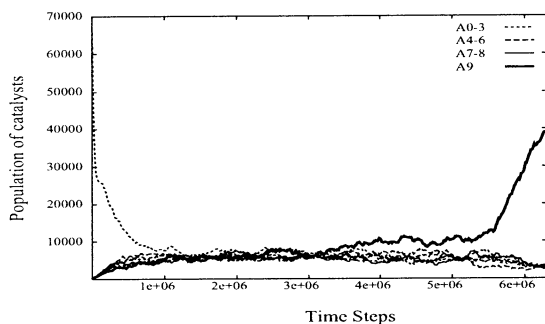
Figure 3: Evolution of catalysts.

4 Conclusion and discussion

We observed that proto-cells can evolve from a pre-cellular metabolic system even where sufficient resources are supplied only in a locally limited region. We can conclude that the limited environment induced the new evolution rather than restraining it. The catalysts that can produce membrane molecules were selected near the margin of the region where the resource is supplied. Proto-cell structures emerged at these marginal regions, and, once proto-cells were organized, they spread into both regions, i.e. where the pre-cellular metabolism was possible and where it was not. They finally took over the catalysts that did not organize proto-cells throughout the environment.

We suppose that such a heterogeneous environment could be found on the primordial earth. For example, the neighborhood of hydrothermal deposits in the sea bottom is a strong candidate. The pre-cellular autocatalysts could evolve near the source where plenty of resources, i.e., molecules with high energy, are constantly supplied. And they would be able to spread further after acquiring proto-cell structures.

It is worth noting that the fitness landscape of the catalysts looks completely different before and after the organization of proto-cells. During the period in which there had been no cells, the evolution mainly depended on the reproduction rate of each self-replicating molecules. In the case presented above, no distinct difference among the catalysts appeared before the organization of the proto-cells. On the other hand, the object of selection became a cellular structure rather than a molecule after the emergence of the proto-cells. It became more important to maintain the cell membranes stably, therefore, the different catalysts became to be selected. A detailed analysis on the dynamics of the transition should be discussed in future studies.

Note that, in this evolution of proto-cells, genetic information is represented by the composition of all molecules held in a cell rather than the set of chromosomes. We presume that the earliest cell had multiple genomes which reproduced themselves independently. This brings another question, "How did the reproduction of the genomes and the division of the cell come to be regulated?" We are considering extending the LAC model by introducing more diverse chemicals in order to answer this question.

References

[1] Jens Breyer, Jörg Ackermann, and John McCaskill. Evolving reaction-diffusion ecosystems with self-assembling structures in thin films. *Artificial Life*, 4:25–40, 1998.

[2] Barry McMullin and Francisco J. Varela. Rediscovering computational autopoiesis. In Phil Husbands and Inman Harvey, editors, *4th European Conference on Artificial Life*, pages 38–47, Brighton, UK, Jul 1997. MIT press.

[3] Naoaki Ono and Takashi Ikegami. Artificial chemistry: Computational studies on the emergence of self-reproducing units. In Jozef Kelemen and Petr Sosík, editors, *In Proceedings of the 6th European Conference on Artificial Life (ECAL'01)*, pages 186–195, Prague,Czech Republic, 2001. Springer.

[4] Pietro Speroni di Fenizio and Wolfgang Banzhaf. Stability of metabolic and balanced organisations. In Jozef Kelemen and Petr Sosík, editors, *In Proceedings of the 6th European Conference on Artificial Life (ECAL'01)*, pages 196–205, Prague,Czech Republic, 2001. Springer.

[5] Francisco J. Varela, Humberto R. Maturana, and R. Uribe. Autopoiesis: The organization of living systems, its characterization and a model. *BioSystems*, 5:187–196, 1974.

Workplace Construction:
A Theoretical Model of Robust Self-Replication in Kinematic Universe

Hiroki Sayama
Department of Human Communication, University of Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan   *sayama@hc.uec.ac.jp*

Abstract: To create an artificial system that demonstrates robust self-replication and evolution in the real physical world is among the grand challenges in artificial life [1]. This question was initiated by von Neumann's theoretical work of self-replicating and evolving automata [2,3]. However, von Neumann's automaton, and all the succeeding models after it, have never been subject to physical implementation until today [4]. The most crucial issue obstructing this challenge is the fragility of the self-replication mechanisms against physical perturbations. As a possible solution for this problem, I present a simple theoretical model to enhance the robustness of the self-replication processes, by introducing an additional subsystem that constructs a "workplace" prior to automaton construction [5]. Workplaces are assumed to be solid structures that can be easily assembled under perturbations and can rigidly hold other components during the construction processes. In this paper, the key ideas of this model are shown using von Neumann's formulations and graphical illustrations.

Keywords: self-replication, self-replicating automata, kinematic model, perturbation, robustness, workplace construction.

1. Von Neumann and Self-Replicating Automata

It is well acknowledged that John von Neumann, a great mathematician / physicist who may be best known as a father of the programmable architecture adopted in today's computers, is also one of the pioneers of artificial life for his seminal work on the self-replicating and evolving automaton [2,3,6]. In his latest years, von Neumann challenged an empirical rule that was believed in engineering disciplines that the complexity of products is always smaller than that of the manufacturing machines that produce them. To grapple with this problem, he developed a theoretical discussion and concluded that a counterexample to this rule could exist if a system is made of the following [2]:

A: a universal constructor that creates any arbitrary structure by referring to a static "description tape".

B: a tape duplicator that makes a copy of the description tape.

C: a controller that reads the description tape and passes the written information to the above A and B and appropriately coordinates their behaviors.

I_{A+B+C}: a description tape that specifies how to construct the entire system itself.

These are symbolically written as

$$A + I_X \rightarrow A + I_X + X, \tag{1}$$

$$B + I_X \rightarrow B + I_X + I_X, \tag{2}$$

$$(A + B + C) + I_X \rightarrow (A + B + C) + I_X + X + I_X, \tag{3}$$

$$\begin{aligned} (A + B + C) &+ I_{A+B+C} \\ &\rightarrow (A + B + C) + I_{A+B+C} \\ &+ (A + B + C) + I_{A+B+C}, \end{aligned} \tag{4}$$

where the last form represents a self-replicating process. More importantly, this model also captures the capability of the evolutionary growth of complexity [7]: If the description tape happens to contain an additional content that does not interfere the correct functioning of $A + B + C$, the system produces a mutated system, possibly of higher complexity than its parent, i.e.,

$$\begin{aligned} (A + B + C) &+ I_{A+B+C+F} \\ &\rightarrow (A + B + C) + I_{A+B+C+F} \\ &+ (A + B + C + F) + I_{A+B+C+F}, \end{aligned} \tag{5}$$

where F is an additional component that happened to emerge in the tape by mutation. The existence of and the relationship between phenotypes ($A + B + C$, $A + B + C + F$) and genotypes (I_{A+B+C}, $I_{A+B+C+F}$) illustrated in this formulation hold a close resemblance with those found in reproduction and variation of real (asexually reproducing) organisms. Later, von Neumann proved that such a counterexample *does* exist, at least theoretically, by implementing his very complex universal constructor in a 29-state 5-neighbor 2D cellular automata space [3]. His idea of self-replicating automata was so profound that it was followed by a number of succeeding studies, which now forms one of the central parts of artificial life [6,8].

Many think of von Neumann's work on self-replicating machines as something different, and even weird, compared to his other (probably more famous) achievements in computer science, game theory, quantum physics, nuclear physics, and so on. However, it is worth pointing out that his idea of universal construction with description tapes has a fundamental correspondence with his another idea of universal computation with program codes stored in memory. These two are actually the same in that a system contains an arbitrary sequence of information inside itself, and provided it has a universal capability, it can imitate the behavior of any arbitrary system described in the sequence. Presumably von Neumann had no substantial

distinction between these two in his mind when he considered these ideas.

There is a practical difference, however, between the universal computer and the universal constructor in terms of the requirement for the robustness of the systems to perturbations. The universal computer does not have to be robust by itself, because the computation theory assumes discrete mathematical entities and their state transition in a noiseless world. The responsibility of realizing such a noiseless condition is primarily on the device manufacturing side and not on the system itself. On the other hand, the universal constructor should be robust *per se*, since its construction capability best makes sense in our real, three-dimensional kinematic universe. Thus the system should be able to deal with more or less continuous physical entities that inevitably involve fluctuations and uncertainty. In such settings, the system itself *is* responsible for the robustness of its workings.

The robust universal construction is apparently a very difficult problem to attain. (Fig. 1) Von Neumann himself tried to consider a kinematic model at first, but later he abandoned it and switched to cellular automata to avoid this difficulty. Although his work was still monumental and stimulative enough even on cellular automata, the lack of robustness was crucial when considered as a model of real biological and/or engineering systems. Therefore, in spite that his universal computer has been implemented hundreds of millions of times (i.e., computers we use nowadays), there has been *no* physical implementation of his universal constructor [4]. So the relevant question to ask here is: *How can we enhance the robustness of self-replication processes?*
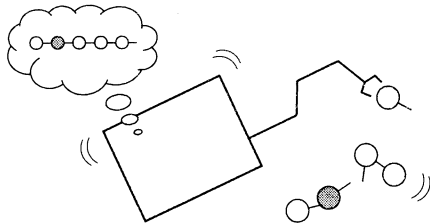
Figure 1: Construction is difficult in kinematic settings.

2. Workplace Construction Model

In this section I present a simple extension of von Neumann's theoretical model, which I tentatively call a *workplace construction model* [5], to illustrate a possible solution for the question mentioned above. The key idea is the introduction of an additional subsystem that constructs a "workplace" prior to automaton construction. This model can be described within von Neumann's framework by putting the added subsystem as a variant of automaton F, except for three additional assumptions I am going to make in what follows.

The first assumption is a reasonable statement that we all know empirically:

Assumption 1: The construction and duplication

processes that are originally sensitive to perturbations can be substantially stabilized by putting them on some solid supporting structure, or *workplace*. (Fig. 2)
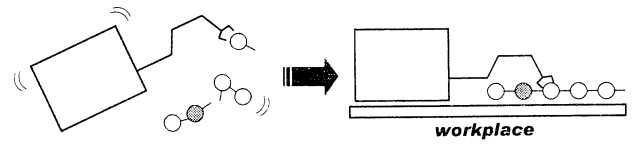
Figure 2: Assumption 1.

For example, drawing a straight line is rather difficult by a free hand but is quite easy and precise by using a ruler. Using this assumption, I symbolically represent such stabilized processes by

$$A + I_X + S^A_X \Rightarrow A + I_X + S^A_X + X, \tag{6}$$

$$B + I_X + S^B_X \Rightarrow B + I_X + S^B_X + I_X, \tag{7}$$

$$(A + B + C) + I_X + S^A_X + S^B_X$$
$$\Rightarrow (A + B + C) + I_X + S^A_X + S^B_X + X + I_X, \tag{8}$$

where S^A_X and S^B_X represent the workplaces that support automaton construction and tape duplication, respectively. Subscript X means that the size and/or shape of the workplaces may depend on the product, if not always. The bold right arrow "$\Rightarrow$" denotes that the process is significantly robust to perturbations, i.e., the outcome is not affected virtually by small but positive amount of perturbations. Here let us keep ourselves quite loose in evaluating the robustness; I just classify processes as either "robust" or "sensitive" to perturbations. Such a simplification enables us to think about the problem more clearly and concisely.

The above forms (6)–(8) look quite similar to the original ones (1)–(3), except for the addition of the two workplaces, so one may want to simply apply $X = (A + B + C) + S^A_X + S^B_X$ to the last form to obtain a robust self-replication process. However, this would result in a non-trivial problem: By definition, workplaces are assumed to hold and support the construction process of the product, so its size in general should be equal to or greater than that of the product. Therefore, if one lets $X = (A + B + C) + S^A_X + S^B_X$, then S^A_X must be large enough to hold the entire product X, while X is positively larger than S^A_X due to the inclusion of other components in it, resulting in a vicious circle that never closes.

Note that this problem is deeply related to the complexity decreasing rule empirically seen in engineering: To obtain a highly organized product, one needs a stabilized and well controlled manufacturing process, which requires a machine that can control the local environment that should be equal to or larger than the product. Part of the reason why von Neumann's machine can seemingly construct a product more complex (larger) than itself is, in this context, because his machine is implicitly supported by an infinite array of cellular automata that virtually works as a solid

workplace. This, however, never applies to our real world where the original complexity decreasing rule came from. In other words, he intended to overcome this empirical rule, but what he actually chose was avoiding it by putting away the crux of the issue out of consideration.

In order to resolve the above problem, here I make the second assumption:

Assumption 2: Workplaces are generally made of a simple but extensive repetition of the same kind of components, so they can be generated with enough preciseness by itself (without another workplace) even under perturbations. (Fig. 3)
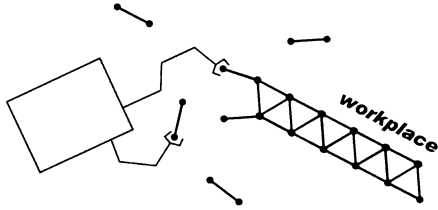
Figure 3: Assumption 2.

Readers may wonder if making this assumption might be just another way of avoiding the problem. Since the main aim of this paper is to show he argument and promote discussions on it, I do not mean this is the one and only right solution. Nonetheless, many empirical observations seem to support this assumption. For example, one can create a straight line in a robust fashion by combining tiny rods into triangular meshes or trusses. As long as the tiny rods are at the same length, the outcome can be precise enough. This is intuitively because the product of this process is not complex; it can be produced by a repetition of the same simple tasks, which is not the case for more sophisticated processes like automaton construction. With this second assumption, I introduce a set of new robust construction processes, i.e.,

$$R^A + I_X \Rightarrow R^A + I_X + S^A_X, \qquad (9)$$

$$R^B + I_X \Rightarrow R^B + I_X + S^B_X, \qquad (10)$$

where R^A and R^B are subsystems that estimate the size of needed workplaces for A and B from the description tape and construct them in a robust fashion. Note that the exact estimation of the size of needed workplaces might be another complex task that needs another workplace to carry it on. We thus have to make the third assumption:

Assumption 3: The size (or the upper bound of the size) of workplaces for construction and duplication processes can be estimated from the description tape in a simple operation, even under perturbations. (Fig. 4)

This assumption is much less obvious than the previous two, and should be subject to discussion and verification. If it would not be the case in some condition, the arguments developed below would lose its universality,
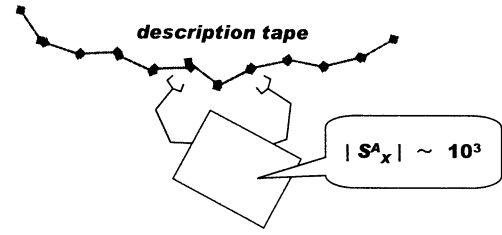
Figure 4: Assumption 3.

but I believe it would still have sufficient implications for how to create robust systems.

Provided the third assumption applies, the previous form of stabilized automaton construction (8) can be rewritten as

$$(A + B + R^A + R^B + C) + I_X$$
$$\Rightarrow (A + B + R^A + R^B + C) + I_X + S^A_X + S^B_X$$
$$\Rightarrow (A + B + R^A + R^B + C) + I_X + S^A_X + S^B_X$$
$$+ X + I_X, \qquad (11)$$

where the entire system $(A + B + R^A + R^B + C)$, which we call G hereafter, is now capable of both constructing X and duplicating I_X in a robust manner, while keeping its own size finite and independent of what is written in I_X. Note that the function of controller C is more complex than before; it now has to coordinate the behaviors of four other subsystems: A, B, R^A, and R^B.

Then, finally, we put the system G itself into X in the above form, to obtain

$$G + I_G \Rightarrow G + I_G + S^A_G + S^B_G$$
$$\Rightarrow G + I_G + S^A_G + S^B_G + G + I_G. \qquad (12)$$

This represents a robust self-replication process of G, which also produces byproducts S^A_G and S^B_G. (Fig. 5)

3. Discussion

The workplace construction model presented above is by no means theoretically or experimentally proven. It uses three assumptions, each of which must be carefully checked about its validity. Whether such a robust automaton can be actually implemented is another key issue to be investigated, probably with a lot of efforts, just like what von Neumann did with cellular automata to show a concrete example of his self-replicating machine.

Despite these problems all, one can obtain a supportive implication for the model from a variety of phenomena observed in real biological systems at various scales. They imply the applicability and effectiveness of the central idea of the presented model. For example, at the smallest level, the formation of cell membranes is probably the most fundamental instance of workplace construction; it isolates the metabolic process of the cell from environmental perturbations and keeps all the sensitive parts at the same place without diffusion. At a much higher level, niche construction seen in ecology [9] is another clear instance of workplace construction, such as dam building by beavers or development of artificial living environment by humans. These controlled environments are literally "workplaces"
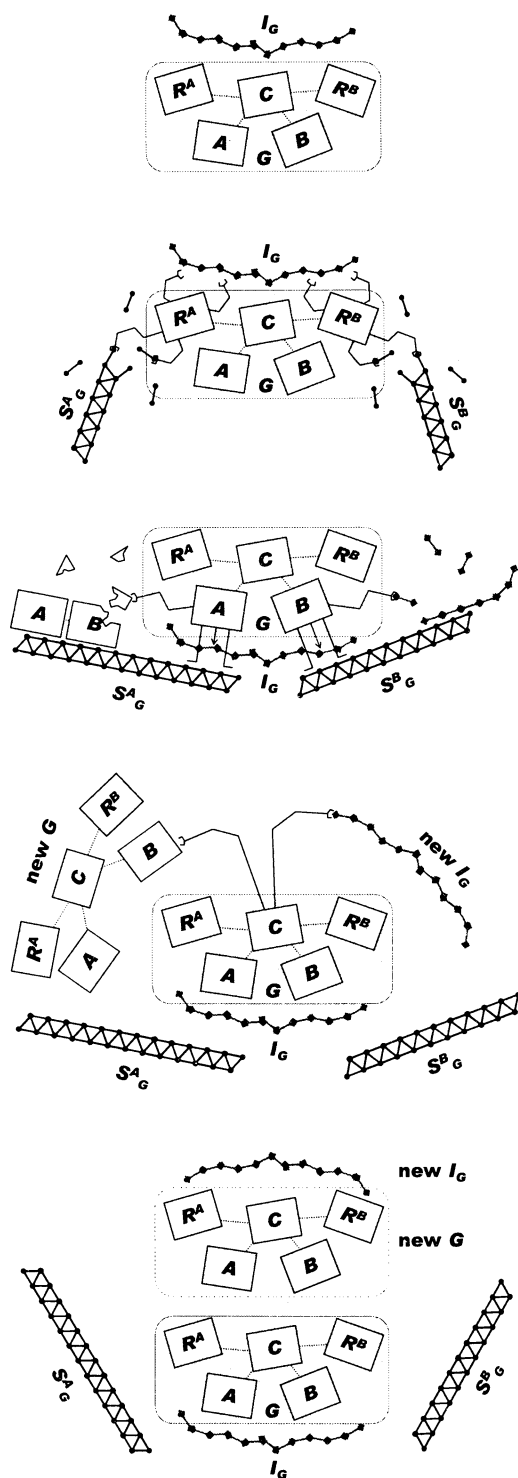
Figure 5: Self-replication with workplace construction.

for their activities.

More directly relevant cases can be found in animals that are about to produce their next generation. For example, firm eggshells of birds and reptiles are probably the most direct example of workplace construction. Although the automaton constructors are embedded in the fetuses in these cases, the purpose of eggshells is exactly the same as is discussed in this paper, i.e., to stabilize the process of offspring construction by

isolating it from the outside and holding it on a solid structure. More sophisticated instance is the uterus of mammals, where the workplace is included in the parent's body, but can be refurbished and extended as needed to hold its offspring under construction. It is also interesting that the production of S^A_G and S^B_G in (12) correctly captures the nature of the process that these workplaces constructed in real organisms (eggshells, endometria, placentas, etc.) are all just for temporary use and they will eventually become garbage after birth of the offspring.

The key conclusion illustrated by the workplace construction model is a simple fact: the more complex a system becomes, the better controlled local environment the system needs in order to construct its replica under perturbations. Biological organisms seem to have evolved such sophisticated "workplaces" for their survival and prosperity. However, this point has been long missing in the earlier studies of artificial life, and it must be taken into account to proceed toward the next step of this interesting research field.

References

[1] M. A. Bedau, J. S. McCaskill, N. H. Packard, S. Rasmussen, C. Adami, D. G. Green, T. Ikegami, K. Kaneko, and T. S. Ray (2000) "Open Problems in Artificial Life", *Artificial Life* 6:363−376.

[2] J. von Neumann (1951) "The General and Logical Theory of Automata", in *Cerebral Mechanisms in Behavior—The Hixon Symposium*, 1−41, John Wiley, New York, NY. Originally presented in 1948.

[3] J. von Neumann (1966) *Theory of Self-Reproducing Automata*, University of Illinois Press, Urbana, IL. Edited and completed by A. W. Burks.

[4] G. Friedman (1996) "The Space Studies Institute View on Self-Replication", *The Assembler: Newsletter of the Molecular Manufacturing Shortcut Group of the National Space Society* 4:4 (http://www.islandone.org/MMSG/9612.html).

[5] H. Sayama (2002) "Von Neumann's Machine in the Shell: Enhancing the Robustness of Self-Replication Processes", in *Artificial Life VIII: Proceedings of the Eighth International Conference on Artificial Life*, in press.

[6] P. Marchal (1998) "John von Neumann: The Founding Father of Artificial Life", *Artificial Life* 4:229−235.

[7] B. McMullin (2000) "John von Neumann and the Evolutionary Growth of Complexity: Looking Backwards, Looking Forwards...", in *Artificial Life VII: Proceedings of the Seventh International Conference on Artificial Life*, 467−476, MIT Press.

[8] M. Sipper (1998) "Fifty Years of Research on Self-Replication: An Overview", *Artificial Life* 4:237−257.

[9] K. N. Laland, J. Odling-Smee, and M. W. Feldman (2000) "Niche Construction, Biological Evolution, and Cultural Change", *Behav. Brain Sci.* 23:131−175.

Self-reproduction and shape formation in two and three dimensional cellular automata with conservative constraints

Katsunobu IMAI[†], Yousuke KASAI[†], Yuya SONOYAMA[‡],
Chuzo IWAMOTO[†], and Kenichi MORITA[†]

[†]Faculty of Engineering
Hiroshima University
Higashi-Hiroshima, 739-8527, Japan

[‡]Matsushita Electric Industrial Co., Ltd.
Kadoma, 571-8501, Japan

Abstract

Conservative cellular automata such as reversible cellular automata (RCA) and number-conserving cellular automata (NCCA) can be thought as models of the physical conservation law of mass or energy and they are used for modeling physical phenomena. In this paper, we show that complex functions such as logical universality, self-reproduction, and shape-formation can be realized in RCA and NCCA.

1 introduction

A cellular automaton (CA) is a system of infinite number of identical cells with finite states which are placed uniformly in the space and each cell changes its internal states by communicating with neighboring cells. A reversible cellular automaton (RCA) [1] is a special type of CA. Its global function is injective (one-to-one) and every configuration has at most one predecessor. Intuitively, it "remembers" the initial configuration and one can reconstruct its initial configuration from a configuration of any time. A number-conserving cellular automaton (NCCA) [2] is a CA such that all states of cells are represented by integers and the total number of its configuration is conserved throughout its computing process. They can be thought as a kind of modelization of the physical conservation law of mass or energy and they are widely used for modeling physical phenomena. Although there are a lot of studies about artificial life, such as self-reproducing models, by employing the framework of CA, it is not so many studies are made by CA with such constraints. In the two-dimensional case, there is no algorithm to determine a CA is reversible or not [3] and it is very difficult to design RCAs. But using Partitioned CA [4], it is possible to design a special type of RCA easily and there is a two-dimensional self-reproducing RCA SR_8 [5] and

its three-dimensional version SR_9 [6]. In contrast to reversibility, number-conservation is a local constraint for CAs. So it is fairly easy to handle and there is an efficient algorithm to determine a two-dimensional CA is number-conserving or not [7]. But to design NCCAs with complex transition rules are still difficult. As far as ±45-degree reflection-symmetric von Neumann neighbor CA, there is a method for constructing NCCAs with complex transition rules and logically universal CA and Langton's self-reproducing CA [8] can be embedded into NCCAs [9].

In this paper, in the reversible case, we show that self-reproducing model, SR_9 has rich abilities to shape formation and in the number-conserving case, Serizawa's computation and construction universal CA [10] can be embedded into a NCCA. Because Serizawa's CA is construction universal in von Neumann's sense [11], number-conserving cellular space can be construction universal in this sense.

2 Conservative Cellular Automata (CA)

2.1 Reversible partitioned CA

A partitioned cellular automaton (PCA)[4] with von Neumann neighborhood is regarded as a subclass of standard CA. Each cell is partitioned into the equal number of parts to the neighborhood size (5 parts) (Fig. 1) and the information stored in each part is sent to only one of the neighboring cells. Each transition rule is depicted as in Fig. 1. In PCA, injectivity of global function is equivalent to injectivity of local function, thus a PCA is reversible if its local function is injective [4]. Using PCA, we can construct a reversible CA with ease. In the three-dimensional case, each cell is partitioned into 7 parts and each transition rule is depicted as in Fig. 2.
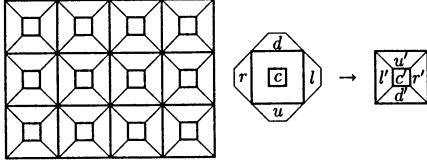
Figure 1: Cellular space and a rule of two-dimensional PCA.
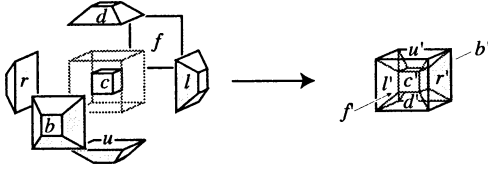
Figure 2: A rule of three-dimensional PCA.

2.2 ±45-degree reflection-symmetric NCCA

If there exists a two-ary function g (called *flow function*) and each transition rule of a von Neumann neighbor CA is represented by the following form, the CA is number-conserving [9].

$$f(c, u, r, d, l) = c + g(c, u) + g(c, r) + g(c, d) + g(c, l),$$
$$g(c, u) = -g(u, c).$$

In the three dimensional case, each transition rule has the following form.

$$f(c, u, r, d, l, f, b) = c + g(c, u) + g(c, r) + g(c, d)$$
$$+ g(c, l) + g(c, f) + g(c, b),$$

Using this condition, we have only to design a two-ary flow function for constructing a NCCA.

3 Self-reproduction in reversible CA

3.1 Three-dimensional self-reproducing RPCA (SR_9)

SR_9 is a three-dimensional self-reproducing RPCA. It has 9 states in each of 7 parts of a cell and it can self-reproduce various structures called "*Worms*" and "*Loops*." Fig.3 is a simple Worm in SR_9. SR_9 has several commands for advancing and twisting its head(Table. 4). For example, when both wires of a Worm have the same sequence 'AB AA AC' (or 'AC AA AB'), its head is twisted leftward (rightward). Using these commands, complex three-dimensional Worms and Loops are constructible.
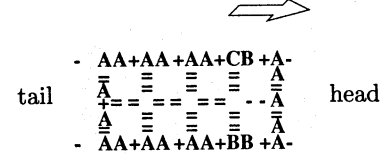
Figure 3: A simple Worm in SR_9.

Figure 4: 'A'-Commands for width 3 shaped worm.

Command				Operations
wire 1		wire 3		
First Signal	Second Signal	First Signal	Second Signal	
A	A	A	A	Advance the head forward
A	B	A	C	Advance the head leftward
A	C	A	B	Advance the head rightward
A	B	A	B	Start rotating (leftward)
A	C	A	C	Start rotating (rightward)

| left turn | right turn | upward turn | downward turn |
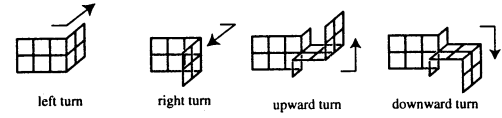
Figure 5: Four kind of turns in SR_9.

3.2 Reflective behavior of loops in SR_9

In this section, we describe that SR_9 has an ability to change shapes of a daughter Loop even if its space is reversible ([12]).

Fig.6 shows a generation example of a different shape of daughter Loop. Although the mother Loop is square, its daughter Loop is rectangle. The length n

Figure 6: Changing the shape of a daughter Loop in SR_9. (t=0,28,100,110)

of this square is 24 and it has the following command sequence instead of AA^{24}.

$$AA^5 ABAA^6 AA^3 ABAAABAA^6$$
$$AA^5 ACAA^6 AA^3 ACAAACAA^6$$

Once a duplication process starts, the shape is encoded as shape signals and they are transmitted along the arm after all positioning signal are transmitted. Thus the following command sequence (length $2n - 2 = 46$) is generated when it is completely expanded into a

Worm.

$$(AA^5ABAA^6AA^3ABAAABAA^6)AA^5ABAA^5ABAA^5ABAA^4$$
$$\underbrace{(AA^5ACAA^6AA^3ACAAACAA^4)}_{\text{positioning signals}}\underbrace{AA^5ACAA^5ACAA^5ACAA^6}_{\text{shape signals}}$$

In normal self-reproducing process of SR_9, daughter Loops are constructed after all positioning commands are used for advancing the constructing arm. But in this case, the daughter Loop is closed soon after its arm are split from the mother Loop. Thus the positioning signals and the shape signals are exchanged.

The next example is more complex one. The diagram of its shape is depicted in Fig.7(a) and it has the following command sequence.

$$\underbrace{\begin{array}{l}(\text{AA AB AA AC AA AC AA AB AA}^4\text{AA}^6\text{AB AA AC AB AA AB AA AC AA AC AB AA AB AA}^7\text{AB AA})\\(\text{AA AB AA AC AA AC AA AB AA}^4\text{AA}^6\text{AC AA AB AC AA AC AA AB AA AB AC AA AC AA}^7\text{AB AA})\end{array}}_{\text{positioning signals}}$$

$$\underbrace{\begin{array}{l}\text{AA AC AA AB AA AB AA AC AA}^2\text{AB AA}^7\text{AB AA}^2\text{AB AA AC AA AC AA AB AA}^2\text{AB AA}^7\\\text{AA AB AA AC AA AC AA AB AA}^2\text{AC AA}^7\text{AC AA}^2\text{AC AA AB AA AB AA AB AA AC AA}^2\text{AC AA}^7\end{array}}_{\text{shape signals}}$$

Each side of this square has specific patterns and they are changed after self-reproduction. Fig.7(b) is generated from Fig.7(a) when start symbol D is provided at the timing of the command sequence shown above. Underlined part is used to form its shape. This shape/positioning command sequence can generate four different Loops in turn.
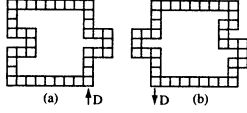
Figure 7: The diagram of changing the shape of a daughter Loop (Period 4) in SR_9.

4 Universal Number-conserving cellular automata

In this section, we show that Serizawa's computation and construction universal CA [10] can be embedded to NCCA.

Serizawa's model uses a *pulse* pattern as signal carriers and two static patterns, a *block* and a *splitter* and 11 actions (A to K) between them are defined. To achieve logical universality, only three actions, "Action A: pulse termination", "Action C: pulse duplication", and "Action H: elimination" are enough. All the other actions are used for achieving construction universality of it. So we first show a logical universal NCCA with former three actions and next we also show an NCCA with all actions.

4.1 Logically universal NCCA

We constructed a logically universal NCCA NC_{15} [13] 15 states (i.e., which uses numbers from 0 to 14) and its quiescent state is 6. The flow function f of NC_{15} has the following values and satisfies $f(a,b) = -f(b,a)$ for all $a,b \in N_{[0,14]}$. Values not defined by these conditions are 0.

$$
\begin{array}{llll}
f(0,6)=5, & f(0,13)=7, & f(1,8)=1, & f(1,11)=5, \\
f(2,0)=2, & f(2,7)=2, & f(4,2)=2, & f(4,9)=1, \\
f(5,3)=1, & f(5,4)=1, & f(5,7)=1, & f(5,8)=1, \\
f(5,10)=1, & f(6,9)=2, & f(6,14)=7, & f(7,12)=3, \\
f(8,2)=1, & f(10,8)=1, & f(10,3)=1, & f(12,3)=1.
\end{array}
$$

NC_{15} has a pulse which advances through the cellular space by changing its shape depicted in Fig. 8. Although Serizawa's pulse has a period 2, a pulse in NC_{15} has a period 3 under the constraint of permutation symmetricity.

Fig.9 shows a block and a splitter in NC_{15}.
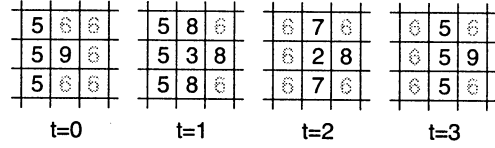
Figure 8: A pulse in NC_{15}.
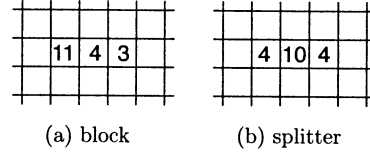
(a) block   (b) splitter

Figure 9: A block and a splitter in NC_{15}.

Fig.12, Fig.11, and Fig.10 are three basic actions collision, duplication, and elimination in NC_{15} respectively.
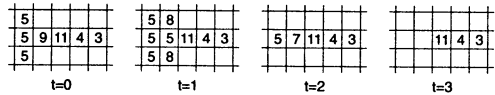
Figure 10: Action A: pulse termination in NC_{15}.

Combining these actions, basic logical elements can be realized and thus NC_{15} is logically universal.

4.2 Construction universal NCCA

In contrast to the Langton's Loop, von Neumann's self-reproducing CA is based on construction universality[11].In the sense of his definition,
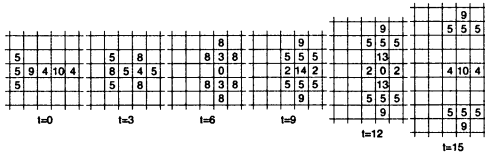
Figure 11: Action C: pulse duplication in NC_{15}.
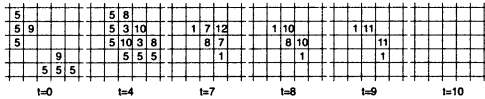
Figure 12: Action H: elimination of two pulses in NC_{15}.

Serizawa's CA is construction universal as well as the Game of Life[14]. We constructed a NCCA $NC_{Serizawa}$ [13] which realizes all 11 actions defined mentioned above. $NC_{Serizawa}$ is construction universal, in the sense that Serizawa's construction universal CA can be embedded into the cellular space. $NC_{Serizawa}$ is defined as follows. $NC_{Serizawa}$ uses numbers from 0 to 110 and its quiescent state is 50.

List of flow functions and several examples of transition processes can be found at the following WWW site.

    http://www.iec.hiroshima-u.ac.jp/
    projects/ncca/pm45/.

5 Summary

In this paper, we show three-dimensional self-reproducing RCA SR_9 has rich abilities of shape formation. It is derived from its self-inspective mechanism. Data as shapes and programs as command sequences are represented in the same manner. We also show Serizawa's logically and construction universal CA can be embedded into NCCAs. Thus NCCA can be construction universal. In the reversible case, the existence of construction universal RCA with a small number of states is still not known.

References

[1] Toffoli, T. and Margolus, N.: *Cellular Automata Machines*, The MIT Press, Massachusetts (1987).

[2] Boccara, N. and Fukś, H.: Number-conserving cellular automaton rules, *Fundamenta Informaticae* **53** (2002) 1–13.

[3] Kari, J.: Reversibility of 2D cellular automata is undecidable, *Physica*, **45D** (1990) 379–385.

[4] Morita, K., and Harao, M.: Computation universality of one-dimensional reversible (injective) cellular automata, *Trans. IEICE Japan*, **E72**, 6 (1989) 758–762.

[5] Morita, K., and Imai, K.: Self-reproduction in a reversible cellular space, *Theoretical Computer Science*, **168** (1996) 337–366.

[6] Imai, K, Hori, T and Morita, K: Self-reproduction in three-dimensional reversible cellular space, *Artificial Life* 8 (2002) 155-174, (http://kelp.iec.hiroshima-u.ac.jp/projects/rca/sr3d/).

[7] Durand, B., Formenti, E. and Roka, Z.: Number conserving cellular automata: from decidability to dynamics, nlin.CG/0102035, (2001) (http://arxiv.org/list/nlin.CG/0102).

[8] Langton, C. G.: Self-reproduction in cellular automata, *Physica*10D(1984) 135-144.

[9] Imai, K, Fujita, K, Iwamoto, C. and Morita, K.: Embedding a logically universal model and a self-reproducing model into number-conserving cellular automata, *Proc. the 3rd International Conference on Unconventional Models of Computation (UMC'02)*, Kobe, LNCS2509, Springer (2002) 164–175.

[10] Serizawa, T.: 3-state Neumann Neighbor cellular automata capable of constructing self-reproducing machine, *Trans. IECE Japan* **J-69-D**, 5 (1986) 653-660 (in Japanese).

[11] von Neumann, J.: *Theory of self-reproducing automata* (ed. A.W.Burks), The University of Illinois Press, Urbana (1966).

[12] Imai, K, Fujita, K, Fori, T. and Morita, K.: Self-reproduction and shape formation in three-dimensional cellular space, *Proc. Cellular Automata Symposium 2001*, Yokohama, (2001) 11–16.

[13] Sonoyama, Y: A study of universality in number-conserving cellular space, Master thesis, Graduate School of Engineering, Hiroshima University (2002) (in Japanese).

[14] Berlekamp, E., Conway, J., Guy, R.: *Winning ways for your mathematical plays*, Vol. 2, Academic Press, New York (1982).

P Automata with Membrane Channels

Marion OSWALD
Department of Computer Science
Vienna University of Technology
Karlsplatz 13, A-1040 Wien
marion@emcc.at

Rudolf FREUND
Department of Computer Science
Vienna University of Technology
Karlsplatz 13, A-1040 Wien
rudi@emcc.at

Abstract

We investigate a variant of purely communicating P systems which are able to accept multisets or even strings given as sequences of terminal symbols taken from the environment. We show that such P automata with membrane channels equipped with only one membrane and specific activating and prohibiting rules can already recognize any recursively enumerable language of multisets and strings, respectively. Moreover, using only activating rules of a very special kind, we obtain a characterization of regular languages.

Keywords: membrane computing, P automata, universality

1 Introduction

Membrane systems were introduced by Gh. Păun in [11] in 1998; since then many variants of membrane systems (then also called P systems) have been investigated, e.g., see [2], [11], [12]; for a comprehensive overview see [13]; the actual status of P systems research can be seen at [9]. One of the most important features of these unconventional models of computation that are abstracted from cell functioning is the *membrane structure* consisting of membranes hierarchically embedded in the outermost *skin membrane*; every membrane encloses a *region* possibly containing other membranes. All the membranes are labelled in a one-to-one manner by natural numbers so that a membrane structure can uniquely be described by a string of correctly matching parentheses (with the skin membrane always labelled by 1), where each pair corresponds to a membrane. In the membrane structure, where membranes act as separators as well as communication channels, multisets of objects can evolve according to given evolution rules, whose non-deterministic, maximally parallel application constitutes a computation. Only halting computations pro-

duce a result, which consists of the objects present in a specified output membrane.

Recently, purely communicating P systems, where the objects only cross membranes without being affected by the rules, have been introduced in [10] and further been investigated in [7]. Formalising biological ways of transporting molecules through membranes in a collaborating manner, objects can cross membranes in the same direction (symport) or in opposite directions (antiport). In the model presented here, objects can cross the membranes by passing through corresponding channels that have been opened by means of activators, unless the channel is blocked by a prohibitor. Applying an activating rule means that an activator multiset (or a single activator symbol) opens input and output channels for specific objects. In the following substep, each object can pass through the surrounding membrane provided there is no prohibitor active, which prevents the object from passing through the corresponding channel. Whereas P systems with activated / prohibited membrane channels used as computing and generating devices have been investigated in [4], we here consider such systems as accepting devices, using them for analysing an input sequence of terminal symbols, as done for the first time in [1]. According to usual notations in formal language theory we will call such devices *P automata with membrane channels*. A similar approach of a purely communicating model of P systems recognizing input strings was considered in [5] under the name of analysing P systems with antiport rules.

2 Preliminary definitions

The set of non-negative integers is denoted by N_0, the set of positive integers by N. An *alphabet* V is a finite non-empty set of abstract *symbols*. Given V, the free monoid generated by V under the operation of concatenation is denoted by V^*; moreover, we define

$V^+ := V^* \setminus \{\lambda\}$, where λ denotes the empty word. A multiset over V is represented as a string over V (and any of its permutations). By $\mid x \mid$ we denote the length of the word x over V as well as the number of elements in the multiset represented by x. We consider two languages L, L' over V to be equal if $L = L'$.

A *deterministic finite automaton* (*DFA* for short) is a quintuple $M = (Q, T_M, \delta, q_0, F)$ where Q is the finite set of *states*, T is the *input alphabet*, $\delta : Q \times T \to Q$ is the *state transition function*, $q_0 \in Q$ is the *starting state* and $F \subseteq Q$ is the set of *final states*. The transition function δ can be extended in a natural way to a function $\delta : Q \times T^+ \to Q$. The language accepted by the DFA M is the set of all strings $w \in T^+$ that are accepted by M in such a way that $\delta(q_0, w) \in F$.

For more notions from the theory of formal languages, the reader is referred to [15].

When considering multisets of symbols, we often take advantage of the universal computational power of register machines (see [8] for some original definitions and [3] for definitions like that we use in this paper).

An *n-register machine* is a construct $RM = (n, R, i, h)$ where n is the number of registers, R is a set of labelled instructions of the form $j : (op(r), k)$ or of the form $j : (op(r), k, l)$, where $op(r)$ is an operation on register r of RM, j, k, l are labels from the set $Lab(RM)$ (which numbers the instructions in a one-to-one manner), i is the initial label, and h is the final label.

The machine is capable of the following instructions:

(A(r),k) Add one to the contents of register r and proceed to instruction k.

(S(r),k,l) If register r is not empty then subtract one from its contents and go to the instruction k, otherwise proceed to instruction l.

HALT Stop the machine. This additional instruction can only be assigned to the final label h.

Such n-register machines can be used to compute any partial recursive function $f : \mathbf{N}_0^k \to \mathbf{N}_0^m$; starting with $(n_1, ..., n_k) \in \mathbf{N}_0$ in registers 1 to k, RM has computed $f(n_1, ..., n_k) = (r_1, ..., r_m)$ if it halts in the final label h with registers 1 to m containing r_1 to r_m. If the final label cannot be reached, $f(n_1, ..., n_k)$ remains undefined.

An n-register machine can also analyse an input $(n_1, ..., n_k) \in \mathbf{N}_0^k$ in registers 1 to k, which is recognized if the register machine finally stops by the halt instruction with all its registers being empty. If the machine does not halt, the analysis was not successful.

From the results proved in [3] (based on the results established in [8]) we immediately conclude the following result:

Proposition 1 *For any recursively enumerable set of vectors of natural numbers $L \subseteq \mathbf{N}_0^k$ there exists a $(k + 2)$-register machine M recognizing L.*

Moreover, for sets of strings we have a similar result (also see [6]):

Proposition 2 *For any recursively enumerable set of strings L over the alphabet T with card $(T) = z - 1$ there exists a 3-register machine M recognizing L in such a way that, for every $w \in T^*$, $w \in L$ if and only if M halts when started with $g_z(w)$ in its first register, where $g_z(w)$ is the z-ary representation of the word w.*

3 P automata with membrane channels

A *P automaton with membrane channels* is a construct $\Pi = (V, T, \mu, w_1, ..., w_n, R_1, ..., R_n)$, where V is an alphabet of *objects*; $T \subseteq V$ is the terminal alphabet; μ is a *membrane structure* (with the membranes labelled by natural numbers $1, ..., n$ in a one-to-one manner); $w_1, ..., w_n$ are multisets over V associated with the regions $1, ..., n$ of μ; $R_1, ..., R_n$ are finite sets of *rules* associated with the compartments $1, ..., n$, which can be of the following forms:

1. *activating rules*: $\langle P; x, out; y, in \rangle$, where $x, y \in V^*$ and P is a finite multiset over V,

2. *prohibiting rules*: $\langle b, out; Q \rangle$ or $\langle b, in; Q \rangle$, where $b \in V$ and Q is a finite multiset over V.

Starting from the *initial configuration*, which consists of μ and $w_1, ..., w_n$, the system passes from one configuration to another one by non-deterministically in a maximally parallel way applying rules from R_i in the following sense: Let $x = x_1...x_m$ and $y = y_1...y_n$. An activating rule $\langle P; x, out; y, in \rangle$ means that by the activator multiset P an output channel for each symbol x_i, $1 \le i \le m$, is activated, and for each y_j, $1 \le j \le n$, an input channel is activated. In the following substep of a derivation (computation), each activated channel allows for the transport of one object x_i and y_j, respectively, provided there is no prohibitor multiset Q active by a prohibiting rule $\langle x_i, out; Q \rangle$ or $\langle y_j, in; Q \rangle$, respectively (which means that the multiset Q can be found in the underlying compartment).

The activating multisets P in the activating rules have to be chosen in a maximally parallel way. A sequence of transitions is called a *computation*; it is *successful*, if and only if it halts (i.e., no rule can be applied anymore). A multiset or a string w over an alphabet T is accepted by the P automaton with membrane channels Π if and only if there is a successful computation of Π such that the (sequence of) terminal symbols taken from the environment is exactly w. (If more than one terminal symbol is taken from the environment in one step then any permutation of these symbols constitutes a valid subword of the input string.)

4 Results

When considering P automata with membrane channels that consist of the simplest membrane structure, i.e., only the skin membrane, and use only singleton activators in in a specific kind of activating rules but no prohibiting rules, we immediately obtain characterisations of regular languages which is shown in the following result:

Theorem 1 *A (string) language L is regular if and only if it can be recognized by a P automaton with with membrane channels $\Pi = (V, T, [_1]_1, w, R)$ having a singleton axiom in only one membrane and using only activating rules of the form $\langle p; p, out; qa, in\rangle$ with $p, q \in V \setminus T$ and $a \in T$.*

Proof (sketch). Let L be a regular (string) language accepted by the DFA $M = (Q, T_M, \delta, q_0, F)$. Then we can construct a P automaton with membrane channels $\Pi = (V, T, [_1]_1, w, R)$ that also recognizes L with $V = Q \cup T_M \cup \{f\}$ where f is a new symbol; $T = T_M$; $w = q_0$; $R = \{\langle p; p, out; qa, in\rangle \mid \delta(p, a) = q\} \cup \{\langle p; p, out; fa, in\rangle \mid \delta(p, a) = q$ for some $q \in F\}$.

On the other hand, if the regular (string) language L is accepted by a P automaton with membrane channels $\Pi = (V, T, [_1]_1, w, R)$, then it can also be accepted by a DFA $M = (Q, T_M, \delta, q_0, F)$ with $Q = V \setminus T$; $T_M = T$; $q_0 = w$; $\delta = \{\delta(p, a) = q \mid \langle p; p, out; qa, in\rangle \in R\}$; $F = \{p \mid$ there exists no $q \in Q$ and no $a \in T$ such that $\langle p; p, out; qa, in\rangle \in R\}$. $\qquad\square$

The main result established in [8] is that the actions of a deterministic Turing machine can be simulated by a 2-register machine. Based on this result and the proof techniques used, e.g., in [3] and [6], we can immediately show the following results using Proposition 2.

Theorem 2 *Let $L \subseteq T^*$ be a recursively enumerable set. Then L can be recognized by a P automaton with*

membrane channels in only one membrane using only singleton activators and prohibitors.

Proof (sketch). According to Proposition 2, we only have to elaborate how we can read the input string w, generate the encoding $g_z(w)$ and then how to simulate the instructions of a 3-register machine; in fact, the main emphasis lies on the simulation of an n-register machine:

1. An Add-instruction $j : (A(i), k)$ is simulated by the activating rule $\langle j; j, out; ka_i, in\rangle$.

2. A conditional Subtract-instruction $j : (S(i), k, l)$ is simulated by the following rules:

   $\langle j; ja_i, out; kf_j, in\rangle \qquad \langle f_j, in; a_i\rangle$
   $\langle j; j, out; j'j'', in\rangle \qquad \langle j'', in; a_i\rangle$
   $\langle j'; j'j'', out; lf_j, in\rangle \qquad \langle f_j, in; j''\rangle$
   $\langle f_j; f_j, out; f_j, in\rangle$

   The construction of the rules in the P automaton with membrane channels guarantees that rules sending out another object together with the activating symbol can only be used without introducing a failure symbol (trap symbol) if also this other object is present in the skin membrane.

3. The halting instruction $h : HALT$ is simulated by the rule $\langle h; h, out; \lambda, in\rangle$.

Now let us start with the singleton q in the initial configuration. For every $a \in T$ we take $\langle q; q, out; q_a a, in\rangle$. Let us assume we have represented the encoding of the input sequence v taken in so far by $g_z(v)$ symbols A. The encoding of $g_z(va)$ obviously is given by $z * g_z(v) + g_z(a)$. This encoding step is acomplished by the following subprogram of a register machine; its first part represents the multiplication by z:

$$\begin{aligned}
q_a : &\quad (S(1), q_{a,1}, q'_a)\\
q_{a,i} : &\quad (A(2), q_{a,i+1}) \text{ for } 1 \leq i < z\\
q_{a,z} : &\quad (A(2), q_a)\\
q'_a : &\quad (S(2), q'_{a,1}, q''_{a,1})\\
q'_{a,1} : &\quad (A(1), q'_a)
\end{aligned}$$

Now let $k = g_z(a)$; then we finish with the following instruction:

$$q''_{a,i} : \quad (A(1), q''_{a,i+1}) \text{ for } 1 \leq i \leq k - 1$$

The input of the next terminal symbol starts with the activating rule $\left\langle q''_{a,k}; q''_{a,k} a, out; q, in\right\rangle$.

Obviously, the instructions of the subprogram above can be translated into activating and prohibiting rules as already elaborated at the beginning of the proof. The numbers of symbols A and B, respectively, correspond with the contents of registers 1 and 2, respectively.

If no further input symbols should be taken in, we have to compute $2^{g_z(w)}$ before using the following activating rule to start the simulation of the 3-register machine indicated in Proposition 2:

$$\langle q; q, out; q_0, in \rangle$$

where q_0 corresponds with the initial label of the register machine.

Obviously, the P automaton halts if and only if the register machine accepts the input $g_z(w)$. $\qquad\square$

The string to be accepted is given by the sequence of terminal symbols a taken from the environment by activating rules of the form $\langle q; q, out; q_a a, in \rangle$. Obviously, this string can also be interpreted as a representation of the corresponding multiset (or the corresponding vector of natural numbers, respectively), which establishes results similar to Theorem 2 for recursively enumerable multisets over T (and the corresponding sets of vectors of natural numbers, respectively).

5   Conclusion

We have investigated P automata with membrane channels which surprisingly already obtain their maximal recognizing power with the simplest membrane structure using only singleton activators and prohibitors. A very restricted variant using only special activating rules with a singleton activator allows for the characterization of regular languages.

Acknowledgements

References

[1] E. Csuhaj-Varjú and G. Vaszil, P automata, in [14], pp. 177–192.

[2] J. Dassow and Gh. Păun, On the power of membrane computing, *Journal of Universal Computer Science* **5**, 2 (1999), pp. 33–49 (http://www.iicm.edu/jucs).

[3] R. Freund and M. Oswald, Generalized P systems with forbidding context, *Fundamenta Informaticae* **49**, 1-3 (2002), pp. 81–102.

[4] R. Freund and M. Oswald, P systems with activated/prohibited membrane channels, *Membrane Computing 2002* (Gh. Paun, G. Rozenberg, A. Salomaa, C. Zandron, eds.), Lecture Notes in Computer Science, Springer, Berlin (2002).

[5] R. Freund and M. Oswald, A short note on analysing P systems with antiport rules, *Bulletin EATCS*, **78** (2002), pp. 231–236.

[6] R. Freund and Gh. Păun, On the number of nonterminal symbols in graph-controlled, programmed and matrix grammars, *Proc. Conf. Universal Machines and Computations*, Chişinău, 2001 (M. Margenstern and Y. Rogozhin, eds.), Springer-Verlag, Berlin (2001).

[7] P. Frisco and H. J. Hoogeboom, Simulating counter automata by P systems with symport/antiport, in [14], pp. 237–248.

[8] M. L. Minsky, *Computation: Finite and Infinite Machines*, Prentice Hall, Englewood Cliffs, New Jersey, USA (1967).

[9] The P Systems Web Page. http://psystems.disco.unimib.it

[10] A. Păun and Gh. Păun, The Power of Communication: P Systems with Symport/Antiport, *New Generation Computing*, **20**, 3 (2002), pp. 295–306.

[11] Gh. Păun, Computing with Membranes, *Journal of Computer and System Sciences* **61**, 1 (2000), pp. 108–143.

[12] Gh. Păun, Computing with Membranes: An Introduction, *Bulletin EATCS* **67** (1999), pp. 139–152.

[13] Gh. Păun, *Membrane Computing - An Introduction*, Springer-Verlag, Berlin (2002).

[14] Gh. Păun and C. Zandron (eds.), *Pre-Proceedings of Workshop on Membrane Computing (WMC-CdeA2002)*, Curtea de Argeş, Romania (2002).

[15] Rozenberg, G., Salomaa, A. (eds.), *Handbook of Formal Languages*. Springer-Verlag, Berlin (1997)

A language for Artificial Life: A theory and an Implementation of a Parameterized 0L System Programming Language

Taishin Y. Nishida*
Faculty of Engineering
Toyama Prefectural University
Kosugi-machi, Toyama 939-0398, Japan

Abstract

Interactionless Lindenmayer systems, or 0L systems for short, are used as formal models for describing growth of (artificial or natural) plants, (artificial or natural) landscapes, fractal structures, etc. In this paper, a theory of parameterized 0L systems, which are one of most useful class for modelling, is explored. And an implementation of a parameterized 0L system programming language is proposed.

1 Introduction

Lindenmayer systems, or L systems for short, were introduced by A. Lindenmayer for modelling growth of living things, especially plants and alga [1, 2]. There are huge numbers of varieties of L systems. Among them, parameterized 0L systems are most frequently used as modelling tools for describing growth of plants, ecosystems, landscapes, and so on [3, 4, 5, 6]

The basic components of parameterized 0L systems are parameterized letters which have the form $a(x_1,\ldots,x_n)$ where a is a symbol and $x_1,\ldots,x_n$ are real numbers. A sequence of parameterized letters, called a parameterized word, is generated by a parameterized 0L system. Rules of the next form determine generations of parameterized words:

$$\langle predecessor\rangle : \langle condition\rangle \to \langle successor\rangle,$$

where $\langle predecessor\rangle$ is a letter with formal parameters, $\langle condition\rangle$ is a logical expression over the formal parameters, and $\langle successor\rangle$ is a sequence of letters of the form $b(e_1,\ldots,e_m)$ in which b is a symbol and $e_1,\ldots,e_m$ are arithmetic expressions over the formal parameters. A parameterized letter $a(x_1,\ldots,x_n)$ generates a parameterized word w if there is a rule whose $\langle predecessor\rangle$ has the symbol a and n formal parameters, $\langle condition\rangle$ is true under the replacement of every

*Email: nishida@pu-toyama.ac.jp

occurrence of formal parameters with corresponding actual parameters $x_1,\ldots,x_n$, and w is the word which is obtained from the $\langle successor\rangle$ by assigning actual parameters to the formal parameters in the arithmetic expressions. A parameterized word $a_1(\ldots)\cdots a_l(\ldots)$, where $a_i(\ldots)$ is a parameterized letter $(i = 1,\ldots,l)$, generates a parameterized word $w_1\cdots w_l$ if and only if every $a_i(\ldots)$ generates w_i for $i = 1,\ldots,l$.

Although the above story is a summary of description of parameterized 0L systems (cf. [3]), one may think that the definition of parameterized 0L systems is very complicated and lacking in mathematical preciseness. For example, the difference between formal and actual parameters and that between logical and arithmetic expressions are unclear. On the other hand, T. W. Chien and H. Jürgensen have given mathematically complete definition to parameterized 0L systems and they have proved that parameterized 0L systems have computational universal power [7]. But I think that their definition is too rigorous to use for modelling.

In this paper we investigate a concise formal definition of parameterized 0L systems. First we formulate parameterized letters whose parameters belong to a set of real numbers and conditions over parameters from the arithmetic operations and order relations only. Then we define rules over parameterized letters defined here.

P. Prusinkiewicz and J. Hanan have proposed a programming language based on parameterized 0L systems [8]. The formal definition given in this paper makes a theoretical background of the parameterized 0L system programming language. We specify and implement a new parameterized 0L system programming language.

We assume that the reader is familiar with the rudiments of formal language theory (see, e.g., [9]) and the theory of L systems (see [10]).

2 The definition of parameterized 0L systems

In this section we establish the notion of rules over parameterized letters. And then the definition of parameterized 0L systems is explored.

First of all, we clarify notations of parameters and operations on them. Let $\mathbf{R}$ be the set of real numbers and let n be a positive integer. Let $\mathcal{A}(n)$ denote the set

$$\mathcal{A}(n) = \mathbf{R} \cup \mathbf{R}^2 \cup \cdots \cup \mathbf{R}^n.$$

We define operations $+, -, \cdot$, and $/$ over $\mathcal{A}(n)$ as follows: for $x = (x_1, \ldots, x_k)$, $y = (y_1, \ldots, y_l) \in \mathcal{A}(n)$

$$x \operatorname{op} y = (x_1 \operatorname{op} y_1, \ldots, x_m \operatorname{op} y_m)$$

where $m = \min(k, l)$ and op is one of $+, -, \cdot$, or $/$. Let $\{T, F\}$ be a new set and let x be a fixed variable symbol.

Definition 1 Finitely calculable expressions *(FCE for short)* over $(\mathcal{A}(n), x)$ are defined as follows:

1. For every $a \in \mathcal{A}(n)$ a is an FCE.

2. The variable symbol x is an FCE.

3. (a) For an FCE e and every $i = 1, \ldots, n$, $\pi_i(e)$ is an FCE. π_i is called the i-th projection.

   (b) For FCEs e_1 and e_2, $(e_1)\natural(e_2)$ is an FCE. $\natural$ is called flattening.

   (c) For FCEs e_1 and e_2, $(e_1) \operatorname{op} (e_2)$ is an FCE where op is one of $\{+, -, \cdot, /\}$.

4. Nothing else is an FCE.

The set of all FCEs over $(\mathcal{A}(n), x)$ is denoted by $\mathcal{E}(\mathcal{A}(n), x)$. For $c \in \mathcal{A}(n)$, $x \leftarrow c$ is called an *assignment*. An FCE e has a value $e[x \leftarrow c] \in \mathcal{A}(n)$ under an assignment $x \leftarrow c$ which is defined by

1. If $e = a \in \mathcal{A}(n)$, then $e[x \leftarrow c] = a$.

2. If $e = x$, then $e[x \leftarrow c] = c$.

3. (a) If e has a value $v = e[x \leftarrow c] = (v_1, \ldots, v_m)$, then

$$(\pi_i(e))[x \leftarrow c] = \begin{cases} v_i & \text{if } 1 \leq i \leq m \\ v_m & \text{otherwise} \end{cases}.$$

   (b) If e_1 and e_2 have values $e_1[x \leftarrow c] = (v_1, \ldots, v_m)$ and $e_2[x \leftarrow c] = (v'_1, \ldots, v'_{m'})$, respectively, then

$$((e_1)\natural(e_2))[x \leftarrow c] = \begin{cases} (v_1, \ldots, v_m, v'_1, \ldots, v'_{m'}) \\ \quad \text{if } m + m' \leq n \\ (v_1, \ldots, v_m, v'_1, \ldots, v'_{n-m}) \\ \quad \text{otherwise} \end{cases}$$

(c) If e_1 and e_2 have values $v = e_1[x \leftarrow c]$ and $v' = e_2[x \leftarrow c]$, respectively, then

$$((e_1) \operatorname{op} (e_2))[x \leftarrow c] = v \operatorname{op} v'.$$

where op is one of $\{+, -, \cdot, /\}$.

Definition 2 Fundamental predicates *(FP for short)* over $(\mathcal{A}(n), x)$ are defined as follows:

1. T and F are FPs.

2. $x \leq c$ and $c \leq x$ are FPs in which $c \in \mathcal{A}(n)$ and $\leq$ is the order relation on $\mathbf{R}$.

3. For FPs p_1 and p_2, $(p_1) \wedge (p_2)$, $(p_1) \vee (p_2)$, and $\neg(p_1)$ are FPs.

4. Nothing else is an FP.

We call an FP of type 2 a basic FP. We treat, for simplicity, the predicates shown below as basic FPs[1]:

$$x < c, \ c < x, \ x \geq c, \ c \geq x, \ x > c, \ c > x.$$

An assignment $x \leftarrow c$ determines a truth value of an FP as follows.

Definition 3 An FP p has a truth value $p[x \leftarrow c] \in \{T, F\}$ under an assignment $x \leftarrow c$ which is defined by

1. If p is T or F, then $p[x \leftarrow c] = p$.

2. If p is a basic proposition $x \operatorname{op} c'$ *(resp. $c' \operatorname{op} x$)* where op is one of $\{\leq, <, \geq, >\}$, then $p[x \leftarrow c] = T$ if and only if

   (a) $c = (y_1, \ldots, y_m)$, $c' = (y'_1, \ldots, y'_{m'})$, and $m = m'$; and

   (b) $y_i \operatorname{op} y'_i$ *(resp $y'_i \operatorname{op} y_i$)* for $i = 1, \ldots, m$

   hold. Otherwise $p[x \leftarrow c] = F$.

3. For FPs p_1 and p_2 which have some truth values, the truth values for $(p_1) \wedge (p_2)$, $(p_1) \vee (p_2)$, and $\neg(p_1)$ are given in Table 1 *(as usual)*.

Let Σ be a finite alphabet. An element of $\Sigma \times \mathcal{A}(n)$ is said to be a *parameterized letter*. We denote a parameterized letter by $a(c_1, \ldots, c_k)$ instead of $(a, c_1, \ldots, c_k)$. A sequence of parameterized letter of finite length (including length zero) is called a *parameterized word*.

Now we define parameterized rules.

---

[1] We exclude $=$ and $\neq$ from basic FPs. They are expressed as $(x \leq c) \wedge (c \leq x)$ and $(x < c) \vee (c < x)$, respectively.

Table 1: Truth values of logical operations.

p_1	p_2	$(p_1) \wedge (p_2)$	$(p_1) \vee (p_2)$	$\neg(p_1)$
T	T	T	T	F
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

Definition 4 *Let Σ be a finite alphabet and x be a variable symbol. A triplet (a, p, w) is said to be an a-rule if $a \in \Sigma$, p is an FP over $(\mathcal{A}(n), x)$, and $w \in (\Sigma \times \mathcal{E}(\mathcal{A}(n), x))^*$. We usually write an a-rule as:*

$$a(x) : p \to w.$$

The above rule produces a parameterized word $w[x \leftarrow c]$ from $a(c)$ if $p[x \leftarrow c] = T$ where

$$w[x \leftarrow c] = b_1(e_1[x \leftarrow c]) \cdots b_l(e_l[x \leftarrow c])$$

provided $w = (b_1, e_1) \cdots (b_l, e_l)$, $b_i \in \Sigma$, and $e_i \in \mathcal{E}(\mathcal{A}(n), x)$ for $i = 1, \ldots, l$. We write $a(c) \Rightarrow w[x \leftarrow c]$ if $w[x \leftarrow c]$ is produced from $a(c)$ by some a-rule.

Then we define a parameterized 0L system.

Definition 5 *A 5-tuple $G = \langle \Sigma, \mathcal{A}(n), x, P, w \rangle$ is said to be a parameterized 0L system where P is a set of a-rules for $a \in \Sigma$ and $w \in (\Sigma \times \mathcal{A}(n))^*$ is the axiom. The parameterized 0L system G defines the generation relation $\Rightarrow_G$ on $(\Sigma \times \mathcal{A}(n))^*$ as follows:*

$$a_1(c_1) \cdots a_l(c_l) \Rightarrow_G w_1[x \leftarrow c_1] \cdots w_l[x \leftarrow c_l]$$

if $a_i(x) : p \to w_i \in P$ and $a_i(c_i) \Rightarrow w_i[x \leftarrow c_i]$ for every $1 \leq i \leq l$. If there is an $a \in \Sigma$ such that every rule $p \in P_a$ does not satisfy $p[x \leftarrow c] = T$, then a is assumed to have a rule $a(x) : \neg(p_1 \vee \cdots \vee p_k) \to a(x)$ where $P_a = \{a(x) : p_1 \to w_1, \ldots, a(x) : p_k \to w_k\}$ is the enumeration of a-rules.

Under the above settings, we can prove the following theorems[2].

Theorem 1 *For every recursively enumerable language L, there is a parameterized 0L system which generates an isomorphic language to L.*

Theorem 2 *There is a language which is not recursively enumerable and generated by a parameterized 0L system.*

---

[2]The proofs of these theorems are not published and found in the URL http://www.comp.pu-toyama.ac.jp/~nishida

3 Specification and implementation of parameterized 0L system programming language

In [8] a C like syntax is proposed for the parameterized 0L system programming language. We borrow the idea here. The parameterized 0L system programming language given here (abbreviated PLL) has several new statements for generating parameterized words and all C statements. We implement PLL by making a translator from PLL source codes to C source codes and a library which consists of functions corresponding to parameterized 0L system actions.

The syntax of essential statements of PLL are shown as follows:

1. $\langle$RULE$\rangle$ ::= PLMODULE $\langle$symbol$\rangle$ (double $\langle$token$\rangle$ [, double $\langle$token$\rangle$]*) $\langle$RULE BODY$\rangle$

2. $\langle$RULE BODY$\rangle$ ::= { [C_statement | $\langle$PRODUCE$\rangle$]*}

3. $\langle$PRODUCE$\rangle$ ::= produce $\langle$SUCCESSOR$\rangle$;

4. $\langle$SUCCESSOR$\rangle$ ::= ε | $\langle$symbol$\rangle$ [(C_expression [, C_expression]*)]

5. $\langle$SETAXIOM$\rangle$ ::= setaxiom $\langle$SUCCESSOR$\rangle$;

6. $\langle$APPLY PRODUCTION$\rangle$ ::= apply_production ;

7. $\langle$FINAL RULE$\rangle$ ::= FINAL_PLMODULE $\langle$PLL LETTER$\rangle$

where $\langle$symbol$\rangle$ and $\langle$token$\rangle$ are ordinary C identifiers and $\langle$PLL LETTER$\rangle$ is a $\langle$symbol$\rangle$ or an element in the set $\{|, \$, [,], \{, \}, ., \tilde{}, \%, !, +, -, \hat{}, \&, \backslash, /, '\}$. The BNF 1 to 4 correspond to all rules whose $\langle$predecessor$\rangle$s have the same $\langle$symbol$\rangle$, i.e., all $\langle$symbol$\rangle$-rules. The $\langle$condition$\rangle$s of the rules are realized using C condition statements, if — else and/or switch; and the $\langle$successor$\rangle$s are described by $\langle$PRODUCE$\rangle$ statements. The $\langle$SETAXIOM$\rangle$ sets the axiom and the $\langle$APPLY PRODUCTION$\rangle$ generates the next parameterized word from the current word using the rules. Some parameterized letters have no rules and so they are not changed. These letters are described by the $\langle$FINAL RULE$\rangle$ statement.

The next is a sample program of PLL which makes a simple tree-like structure shown in Figure 1.

```
FINAL_PLMODULE +
FINAL_PLMODULE -
FINAL_PLMODULE [
FINAL_PLMODULE ]
```

```
FINAL_PLMODULE F

PLMODULE make_branch(double x, double y){
    if(x >= 0.1){
        produce F(x),
        [, +(10.0), make_branch(x * 0.9, y * 0.707), ],
        [, -(25.0), make_branch(x * 0.7, y * 0.707), ];
    }else{
        produce F(x);
    }
}

PLMODULE A(double c, double d){
    produce F(c),
    [, +(10.0), make_branch(c * 0.9, d * 0.707), ],
    [, -(60.0), make_branch(c * 0.7, d * 0.707), ];
}

void main(void){  '
    int i;
    setaxiom A(1.0,1.0);
    for(i=0; i<10; i++){
        applyproduction;
    }
}
```

In the program the FINAL_PLMODLEs are used for drawing 3-dimensional shapes from generated parameterized words (see [3]).

Figure 1: A simple tree-like structure generated by the sample program.

4   Conclusion

A formal definition of parameterized 0L systems and a few theorems derived from it have been mentioned. A parameterized 0L system programming language (PLL) and its implementation have been introduced. The definition gives PLL theoretical background. Because PLL is not a interpreter but a compiler of parameterized 0L systems, generations of PLL are very fast. There are some shortcomings in the PLL specification given here, e.g., it cannot discriminate parameterized letters which have the same symbol and different numbers of parameters. It will be a future work to make more useful parameterized 0L system programming language.

References

[1] A. Lindenmayer (1968), Mathematical models for cellular interactions in development, I. Filaments with one-sided inputs, II. Simple and branching filaments with two-sided inputs, *J. Theor. Biol.* 18:280–315.

[2] A. Lindenmayer (1971), Developmental systems without cellular interactions, their languages and grammars, *J. Theor. Biol.* 30:455–484

[3] P. Prusinkiewicz and A. Lindenmayer (1990), *The Algorithmic Beauty of Plants*, Springer-Verlag, Berlin.

[4] J. Hanan (1997), Virtual plants — integrating architectural and physiological models, *Environmental Modelling & Software*, 12:35–42.

[5] P. Prusinkiewicz (2000), Simulation modeling of plants and plant ecosystems, *CACM* 43:85–93.

[6] Y. Parish and P. Müller (2001), Procedural modeling of cities, ACM SIGGRAPH 2001, Los Angeles CA USA 12–17 Aug., pp. 301–308.

[7] T. W. Chien and H. Jürgensen (1992), Parameterized L systems for modelling: potential and limitations, in G. Rozenberg and A. Salomaa (eds), *Lindenmayer Systems*, Springer-Verlag, Berlin, pp. 213–229.

[8] P. Prusinkiewicz and J. Hanan (1992), L-systems: from formalism to programming languages, in: G. Rozenberg and A. Salomaa (eds), *Lindenmayer Systems*, Springer-Verlag, Berlin, pp. 193–211.

[9] J. E. Hopcroft and J. D. Ullman (1979), *Introduction to Automata Theory, Languages, and Computation*, Addison Wesley, Reading.

[10] G. Rozenberg and A. Salomaa (1980), *The Mathematical Theory of L Systems*, Academic Press, New York.

Computing with Rho Family GTPases:
Operability and Feasibility

Jian-Qin LIU and Katsunori SHIMOHARA

ATR Human Information Science Laboratories,
2-2-2, Hikaridai, "Keihanna Science City", Kyoto 619-0288, Japan
E-mail: {jqliu, katsu}@atr.co.jp

Abstract -- This paper proposes a new molecular computing method based on the signaling pathways regulated by Rho family GTPases of mammalian cells *in situ*. Furthermore, their operability and feasibility are discussed.

Keywords: molecular computing, kinase computing, 3-SAT problem solving.

1. Introduction

"Kinase computing" is a new molecular computing method with Rho family GTPases of mammalian cells, which differs from the Adleman-Lipton paradigm[1,4,6,8,9] of DNA computing and surfaced-based techniques [2,3].
(1) It employs the signaling pathways (in the potential implementation, these refer to the pathways of Rho family GTPases [7]) of cells for "engineered communications"[5].
(2) It can be formalized as a special kind of (hyper)graph rewriting, thus forming "conceptualized pathway object structure" to systematically guarantee the rigorousness of massive parallel computing processes.
The features of the method are:
(1) constructing the pathways in cells (functional proteins and kinases) as computing units regulated by Rho family GTPases for potential implementation;
(2) linear cost both in controlling space and in time;
(3) parallelism theoretically guaranteed by autonomy of (hyper)graph rewriting.
Our aim is to build a molecular computer using *in situ* cells with robustness of a massive parallel computing power.

2. Molecular Computing by Kinases

The encoding of the computing process by kinases is made by phosphorylation-dephosphorylation mechanism in cells. The principle of kinase-based computing is feasible to potential implementation of the wetware form of the molecular computers using signal pathways regulated by kinases, because that the ATP is related to the kinases' effect in the sense of autonomy (i.e., automation) for the whole signaling processes.

(1) Program Codes
The program codes are the Rho family GTPases. The monitoring procedure of program codes is carried out through the regulation of signaling pathways. One subset of the signaling molecules that encode the combined form of the variables is

A_{kl} = {PIP5-kinase, Rhophilin, Rhotekin, PKN, PRK2, citron, citron-kinase, Rho-kinase, MBS, p140mDia, p140Sra-1, Por1, PI3-kinase, S6-kinase, IQGAP, PAKs, MLK3, MEKK4, MRCKs, WASP, N-WASP, and Ack}.

The lower bound of the alphabet set is 2×10^6. Also the programming paradigm derived from our computation model is guaranteed by (hyper)graph rewriting theory for parallel computing.

(2) Regulation
The target molecules for regulating the pathways of Rho family GTPases are {Rho, Rac, Cdc42} corresponding to the subset A_{kl}.

(3) I/O Relationship
A feasible solution for I/O is to detect the signaling proteins by 2D gel electrophoresis, where the following schemes vary in physically manipulations according to the physical and chemical features of the kinases.
Scheme A: the encoding is made by phosphorylation form of the molecules that represent the positive form of the variables. The clauses in the 3-SAT problem made by the pathways are designed directly to select the phosphorylation/dephosphorylation form of candidate solutions, i.e., to sieve the molecules with the above-mentioned form. Here no signal transduction is made in advance.
Scheme B: In order to describe the form of the positive variables corresponding to the ligase (for signaling processes), the phosphorylation is activated by the signal transduction happened through the cell membrane. Because inter-cell communications and intra-cell communications use different signaling molecules, the inverters of cells are necessary and the pathways of cells sieve the transferred molecules that pass the channels regulated by the concerned pathways. These computing processes are reversible under the controlling conditions. The protocol for above-mentioned kinases are made on the operation set of

Q_k = {DISTRIBUTE, FEED, SIEVE, EXTRACT}

where the read out is made by 2D gel electrophoresis.

3. 3-SAT Computation by Signal Transduction Based on Kinases

The computing process by kinases is demonstrated through the applications in the 3-SAT instance.

(1) 3-SAT Problem:

The 3-SAT problem can be described as:

Let $\Psi_i = C_{i1} \vee C_{i2} \vee C_{i3}$ be a clause (i=0,1, ... m), so all clauses {Ψ_i} produce the constraint Ξ with the form $\Psi_1 \wedge \Psi_2 \wedge ... \wedge \Psi_m$. Our task is to find the solutions -- the set of the combinatorial forms of n variables that satisfied Ξ.

(2) The Representation and Operations Using Molecules in Cells:

We define:

$Mo[X_{ij}]$ -- the molecule that represents the positive form of the variable X_{ij}; i.e., Xij = true (1);

$Mo[PO\text{-}X_{ij}]$, i.e., $Mo[X_{ij}]$ -- the molecule that represents the positive form of X_{ij} by attaching the phosphorylation components to $Mo[X_{ij}]$;

PATH $(PO\text{-}X_{ij})$ -- the signaling pathway that can accept the molecule $Mo[PO\text{-}X_{ij}]$;

$Mo[PN\text{-}X_{ij}]$, i.e., $Mo[\neg X_{ij}]$ -- the molecule that represents the negative form of X_{ij} , i.e., $\neg X_{ij}$ by attaching the de-phosphorylation components to $Mo[X_{ij}]$; i.e., Xij = false (0);

PATH $(PN\text{-}X_{ij})$ -- the signaling pathway that can accept the molecule $Mo[PN\text{-}X_{ij}]$; This also can be represented as the notation $PO\text{--}Mo[X_{ij}]$ and $PN\_Mo[X_{ij}]$;

The input of the signaling pathways is {Ψ_i}.

Let C_{ij} (j=1,2,3) be the j-th variable condition in the i-th clause and Ch_1, Ch_2, ... Ch_m be the signaling molecule channel in the membranes of cells, so Chi responses to {$Ch_{i1} \cup Ch_{i2} \cup Ch_{i3}$}. Here, the channel refers to the RECEPTOR molecules in the cells' membrane. The kinases of pathways recognize the molecules encoded as clauses in 3-SAT and activate the corresponding pathways in the cells. We omit the notations of RECEPTOR for channels in this section (assumed they exit in the corresponding pathways).

In the i--th clause, C_{ij} need the pathway that satisfies the condition of C_{ij}. Different channels are assigned to the different variables in clauses, the molecules representing the same variable from different channels will be selected by another signaling pathway by certain threshold, in order to only keep those molecules that can satisfy the condition of all the clauses, which can be denoted as

PATH[contn[Mo[Q]]>Thd].

Above formula refers to the process in which molecule Q can pass through this unit if its concentration "contn" reach the threshold Thd, else it can not do so. Here the pathways can take the form of atomic or active (non-atomic) forms.

(3) An Example

Based on the encoding schemes suggested in the previous subsection, we can apply the operations:
(a) DISTRIBUTE;
(b) FEED;
(c) SIEVE;
(d) EXTRACT;
where the readout is made by 2D gel electrophoresis (Cf. Fig.1). This process mainly consists of two stages (Cf. Fig.2). We take clause set of 3-SAT problem as {[X1 $\cup$ X2 $\cup$ X3], [X1 $\cup$ $\neg$ X2 $\cup$ X3]}. Therefore, the pool of the candidate solutions is set as:

Order	Combinatorial form of molecules
Cobm(0)	X2 = 0, X1 = 0, X0 = 0;
Cobm(1)	X2 = 0, X1 = 0, X0 = 1;
Cobm(2)	X2 = 0, X1 = 1, X0 = 0;
Cobm(3)	X2 = 0, X1 = 1; X0 = 1,
Cobm(4)	X2 = 1, X1 = 0, X0 = 0;
Cobm(5)	X2 = 1, X1 = 0; X0 = 1;
Cobm(6)	X2 = 1, X1 = 1, X0 = 0;
Cobm(7)	X2 = 1, X1 = 1, X0 = 1;

Fig.1 Overview of the approach
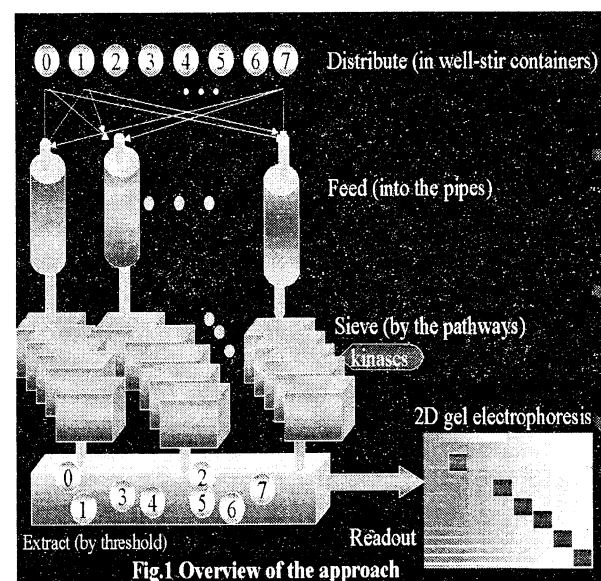
Here the true/false value (i.e., 1/0 for binary logic) refers to T/F, i.e., phosphorylation/dephosphorylation. These are input simultaneously to all the pathways (all the branches) that represent the units:

PATH[X1 ∪ X2 ∪ X3] = PATH[Mol[X1]] + PATH[Mol[X2]] + PATH [Mol[X3]]

Here the "+" refers to the binding of the molecules. we can get that:

Cobm(1), Cobm(2), ..., Cobm(7).

PATH[X1 ∪ not X2 ∪ X3]

Cobm(0), Cobm(1), Cobm(3), ..., Cobm(7).

Fig.2(a) The first stage of the 3-SAT computation

Fig.2((b) The second stage of the 3-SAT computation

Then the PATH W judge the common outputs of PATH[X1 ∪ X2 ∪ X3] and PATH[X1 ∪ not X2 ∪ X3] and give out the set of

{Cobm(1), Cobm(2),...,Cobm(7)}

as the final result. The I/O relationship can be demonstrated in Fig. 2(a) and fig. 2(b). So, the generic modeling derived from our discussion is obvious when

the encoding schemes are extended into any program codes.

As the transductions of the (hyper)graph rewriting, the formal operators of the corresponding implementation consist of the following three major steps:

(1) The output of the pathway for [X1 ∪ X2 ∪ X3] is formalized as:

readout*(PATH(Mo[X1]**Mo[X2]**Mo[X3])
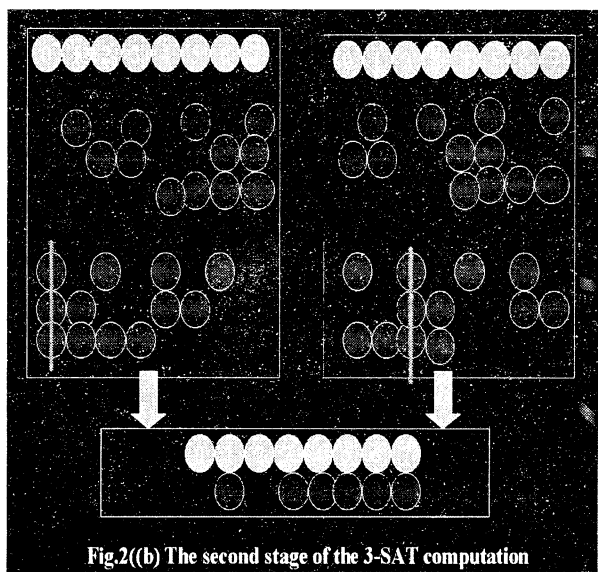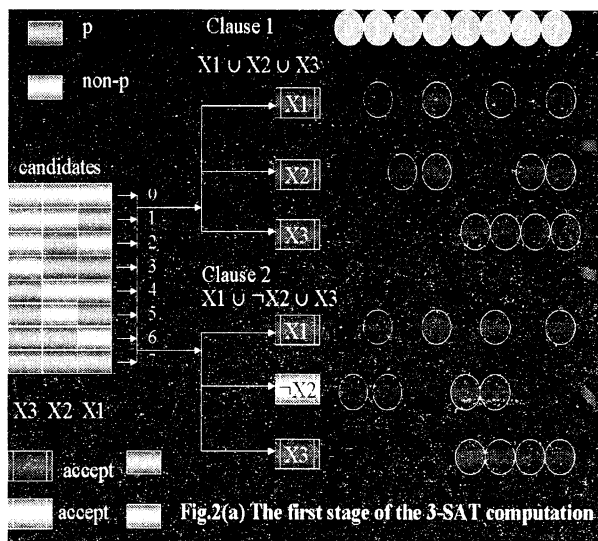
where "readout*(.)" refers to the signal detection at the internal states, i.e., not the final output case, and the states satisfies the combinatorial conditions/sets for the clauses, and the notation "**" refer to the co-existence of different pathways.

(2) The output of the pathway for [X1 ∪ ¬ X2 ∪ X3] is formalized as:

readout*(PATH(Mo[X1]**Mo[X2]**notMo[X3]).

(3) The final result of the whole 3-SAT computation is given as the form:

readout(PATH(common-component-of-pathways
(readout*(PATH(Mo[X1]**Mo[X2]**Mo[X3])))and
(readout*(PATH(Mo[X1]**Mo[X2]**notMo[X3]))))),

i.e., the solution set to the given 3-SAT problem.

4 Preliminary Result
(1) Efficiency:
One of the important tasks in regulatory mechanism of engineered phosphorylation-dephosphorylation processes (circles) is the de-coupling of the phosphorylation-dephosphorylation and controlling of the directed signaling molecular (switch-like in the pathways). The control strategy and techniques need to be developed for guaranteeing the stability of the molecular computation. With the regulation schemes we designed, we have proved that the time complexity and (regulation) space complexity is linear, when the "kinase computing" algorithm is applied to solving the important benchmark -- the 3-SAT problem. We have achieved that

The time complexity is O(1).

The space complexity is O (w) where w = min (m,n).

The result of a brief comparative study is given in Table 1. Here SNT is the Schöning algorithm; SB is the Surface-based DNA computing; SND is the DNA computing implementation of the Schöning algorithm; CRD is the Chen-Ramachandran's algorithm; YSD is the Yoshida-Suyama's algorithm; LSK is the Kinase computing by Liu and Shimohara. "*" refers to the worst case. In "**", α is the parameter in the related dynamics condition.

Comparative Study: the (computational) complexity	Time	Space
SN:	1.33^n	---
SB:	m	2^n
SND:	$kmn+n^3$	$(2-2/k)^n$
CRD*:	k^2mn	$2^{(1-1/k)\cdot n+\log\alpha}$**
YSD:	$n+m$	$2^{0.5n}$
LSK:	m	"$m\times n$ or w" for 2^n
Here $w = \min(m,n)$.		

Table 1: Complexity of six representative algorithms

(2) Error-free Schemes

Theoretically, the molecular computing method we proposed here does not generate the situation about errors in DNA computing. This is because the program codes are made by kinases and related function proteins in cells so that the different codes (e.g., the different variables) are different kinases without any overlaps of the contents under the regulation/controlling of the kinases. Therefore, we are focussing on the regulation schemes of the kinase-based signaling pathways of cells -- the engineered phosphorylation/dephosphorylation of the kinase-based signaling pathways in mammalian cells *in vivo* and *in situ*.

(3) The scalability

Under the condition that the number of different symbols is 2×10^6, an individual cell with the kinase-based pathways can carry out the (2×10^6, int($2/3\times10^6$)) 3-SAT computation. If we use the multi-cells *in situ* to make the whole system of molecular computing, it is obvious that no upper bound of the scale of the 3-SAT computation exist. The important technical issues still will be the regulations for the related cells.

5 Conclusion

We have studied the opearbility of "kinase computing" using Rho family of GTPases and found the merits of efficiency and scalability while the regulation/controlling schmes act as an important factor to guarantee the potential molecular computers free of errors. The encoding schemes of molecular computing by phosphorylation/dephosphorylation are the funmctional switching mechanism for general purpose (of molecular computers). By software simulation, we have confirmed that the potential prototype of kinase computing is feasible for future implementation under the related technical conditions of the regulation of engineered kinase-based signaling pathways of cells.

References

[1] Adleman, L.M., Molecular computation of solutions to combinatorial problems, Science, 266, 1994; 1021-1024.

[2] Mitsunori Ogihara and Animesh Ray, DNA computing on a chip, Nature, 403; 143-144.

[3] Qinghua Liu, Liman Wang, Anthony G. Frutos, Anne E. Condon , Robert M. Corn and Lloyd M. Smith, DNA computing on surfaces, Nature, 403; 175-179.

[4] Masami Hagiya, Perspectives on molecular computing, New Generation Computing, 17, 1999; 131-151.

[5] Weiss, R. and Knight, Jr., T.F., Engineered communications for microbial robotics. In: Anna Condon and Grzegorz Rozenberg (editors), Preliminary Proceedings of DNA6 - Sixth International Meeting on DNA Based Computers, Leiden, The Netherlands, June 13-17, 2000, pp.5-19.

[6] Kensuku Sakamoto, Hidetaka Gouzu, Ken Komiya, Daisuke Kiya, Shigeyuki Yokoyama, Takashi Yokomori and Masami Hagiya, Molecular Computation by DNA Hairpin Formation, Science, Vol.288, 2000; 1223-1226.

[7] K. Kaibuchi, S. Kuroda and M. Amano, Regulation of the cytoskeleton and cell adhesion by the Rho family GTPases in mammalian cells, Annu.Rev.Biochem. 68, 1999; 459-486.

[8] R. Lipton, DNA solutions of hard computational problems, Science, 268, 1995; pp.542-545.

[9] Ravinderjit S. Braich, Cliff Johnson, Paul W.K. Rothemund, Darryl Hwang, Nickolas Chelyapov, and Leonard M. Adleman, Solution of a satisfiability problem on a gel-based DNA computer, In Pre-proc. of DNA 6, 2000; 31-42.

Dynamic Cooperation Control for a Mobile Manipulator

Ko JaePyung, Jin TaeSeok and Lee JangMyung

Intelligent Robot Lab., Dept. of Electronics Eng., Pusan Nat. Univ.
30, Changjeon-dong, Kumjeong-Ku, Pusan 609-735, Korea
Tel : +82-51-510-1696, Fax : +82-51-510-5190, http://robotics.ee.pusan.ac.kr/
E-mail :jmlee@pusan.ac.kr

Abstract

A mobile manipulator is a very useful system to achieve various task in dangerous environment. In this paper, a mobile manipulator is virtually divided into a mobile robot and a task robot, and all the tasks are also divided into task segments that can be performed by only the task robot. An optimal configuration of the task robot is defined by the task-oriented manipulability measure (TOMM) value for a given task segment, which determines the base of the task robot as well as provides the position of the mobile robot at the same time. Assuming the task segment can be performed by the task robot, the objective of the mobile robot is to carry the task robot to the best position and orientation for the given task segment. A sequence of this alternating task execution scheme enables the mobile manipulator to execute various tasks efficiently. The proposed algorithm is experimentally verified and discussed with a mobile manipulator, PURL-II.

1. Introduction

While there has been a lot of work done on the control for both mobile robot navigation and the fixed manipulator motion, there are few reports on the cooperative control for a robot with movement and manipulation ability[1]. And at present, there has been a lot of research on the redundant robot that has more degrees of freedom than the necessary in the given workspace, so it can have optimal position and optimized job performance [2,3]. It is desirable to look at improvement in two areas, one is a case where multiple manipulators cooperate and perform the task in parallel, the other case is where a mobile robot and a task robot perform the task in a sequence. In this paper, we define a mobile manipulator as a mobile robot combined with a vertical multi-joint robot and define a vertical multi-joint robot as a task robot. Being different from the fixed redundant robot, the mobile manipulator has the characteristic that, with respect to the given working environments, has the merits of singularity avoidance, collision avoidance, efficient application of the corresponding mechanical parts and improvement of adjustment. Because of these characteristics, it is desirable that one uses the mobile manipulator with the transportation and dexterous handling abilities in difficult working environments. This paper utilizes the mobile manipulator PURL-II that is a combination of a mobile robot that has 3 degrees of freedom and a task robot that has 5 degrees of freedom for efficient job accomplishment. We have analyzed the kinematics and inverse kinematics of each robot defining the mobile robot and task robot as two parts of the mobile manipulator. The results that we acquired by implementing the proposed algorithm through computer simulation and the experiment using PURL-II are demonstrated.

2. Mobile Manipulator

2.1. Configuration of the Mobile Manipulator

The target robot is an autonomous mobile manipulator, named "PURL-II" which is being developed by our research group. Fig. 1 shows a photograph of the robot. The robot PURL-II consists of the task robot with 5 degrees of freedom and the mobile robot with 3 degrees of freedom. The mobile robot has three DC servomotors for three different directions of movement. We used an 87C196CA microprocessor as the motor controller, so that we could control the three motors concurrently. We mounted the ROB3 with 5 degrees of freedom as the task robot, and installed a gripper at the end-effector so it can grip objects.

Fig. 1. Complete PURL-II.

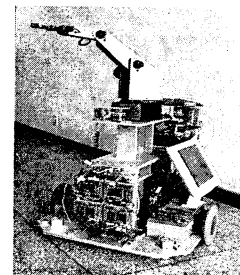## 3. Dynamic Cooperation Control

3.1 Task Segment

Tasks given to a robot can be described by Artificial Constraints. Artificial constraints are represented as the controllable (motion / force) components in the given task environment.

Generally a task can be described as the trajectory of motion and force $w.~r.~t$ the base frame. If the task is described by the relevant task frame selected, the

description can be simplified. By describing the task $w. r.$ t the task frame, we can classify each task's components into task essential, task dispensable and task free component, and consequently can make the task be described by only task essential components. Now, by using the redundancy from the reduced dimensional description of task, $i. e.$, from the description of tasks in terms of only task essential components, the task execution can be optimized. For this, one task is divided into a series of subtasks composed of the same task essential components, and each subtask is defined to as a task segment. When the same task essential components are controlled, that is, a task segment is required, the configuration of the redundant manipulator can be optimized for the task segment.

3.2 Cost Function

In this paper, we have the objective of movement-minimization of the whole robot in performing the task, so we express the vector for mobile manipulator states as

$$s = \begin{bmatrix} p \\ q_t \end{bmatrix} \quad \text{where} \quad p = \begin{bmatrix} x & y & z & \theta \end{bmatrix}^T \quad \text{and} \quad (1)$$

$$q_t = \begin{bmatrix} q_{t1} & q_{t2} & \cdots & q_{tn} \end{bmatrix}^T$$

Here, s is the vector for the robot and consists of p representing the position and direction of mobile robot in Cartesian space and q_t representing the n joint variables of the task robot. Now to plan the task to minimize the whole movement of mobile manipulators, a cost function, L, is defined as

$$L = \Delta s^T \Delta s = (s_f - s_i)^T (s_f - s_i)$$
$$= (p_f - p_i)^T (p_f - p_i) + (q_{tf} - q_{ti})^T (q_{tf} - q_{ti}) \quad (2)$$

Here, $s_i = \begin{bmatrix} p_i & q_{ti} \end{bmatrix}^T$ represents the initial states of the mobile manipulator, and $s_f = \begin{bmatrix} p_f & q_{tf} \end{bmatrix}^T$ represents the final states after having accomplished the task. In the final states, the end-effector of the task robot must be placed at the desired position x_d.

We can express the final position of the mobile manipulator x_f as the function of the desired coordinate x_d, joint variables q_m and q_{tf}, then the cost function that represents the robot movement is expressed as the n*1 space function of q_m and q_{tf} as

$$L = \{x_d - R(q_m)f(q_{tf}) - x_i\}^T \{x_d - R(q_m)f(q_{tf}) - x_i\}$$
$$+ \{q_{tf} - q_{ti}\}^T \{q_{tf} - q_{ti}\} \quad (3)$$

In (3), q_m and q_{tf} which minimize the cost function L must satisfy the condition,

$$\nabla L = \begin{bmatrix} \dfrac{\partial L}{\partial q_m} \\ \dfrac{\partial L}{\partial q_{tf}} \end{bmatrix} = 0 \quad (4)$$

Because the cost function is nonlinear, it is difficult to find analytically the optimum solution that satisfies (4). So in this papers, we find the solution numerically using the gradient method described as

$$\begin{bmatrix} q_{m(k+1)} \\ q_{tf(k+1)} \end{bmatrix} = \begin{bmatrix} q_{m(k)} \\ q_{tf(k)} \end{bmatrix} - \eta \nabla L \Big|_{q_{m(k)}, q_{tf(k)}} \quad (5)$$

This recursive process will stop, when $\|\nabla L\| < \varepsilon \approx 0$, that is, when $q_{m(k)}$ and $q_{tf(k)}$ are optimum.

Through the optimum solutions of q_m and q_{tf}, the final robot state s_f can be calculated as

$$s_f = \begin{bmatrix} p_f \\ q_{tf} \end{bmatrix} = \begin{bmatrix} x_d - R(q_m)f(q_{tf}) \\ q_{tf} \end{bmatrix} \quad (6)$$

3.3 TOMM (Task Oriented Manipulability Measure)

Here we define ^dGME and ^aGME before definition of TOMM[5]. Task Oriented Manipulability Measure (TOMM) represents a measure of shape discrepancy between the desirable manipulability ellipsoid, ^dGME, and the actual manipulability ellipsoid, ^aGME, of a manipulator:

$$TOMM = \sum_{i=1}^{m} (^d\sigma_i - \eta_i)^2 \quad (7)$$

where $^d\sigma_i$ is a desirable length of the principal axis, ^du_i, and $\eta_i = [^du_i^T C^d u_i]^{-1/2}$. ^du_i is the essential direction of motion/force to perform task, which is composed of task essential components. η_i is the length of actual GME obtained along direction, ^du_i.

$^d\sigma_i$ can be determined according to the importance of the control components, motion or force, along the direction, ^du_i. Notice that $^d\sigma_i$ is pre-determined considering task requirements heuristically, as shown in (8). For example, when a large motion is required along ^du_i, $^d\sigma_i$ will be set to large to configure the mobile manipulator suitable for generating the motion.

$$^d\sigma_i = (1-a)\eta_{i,\max} + a\eta_{i,\min} \quad (where \; 0 \le a \le 1) \quad (8)$$

3.4 Dynamic Cooperation

For example, to draw a big rectangle on the wall, the task robot may reach the position out of its workspace. And then, the task robot cannot continue the task unless

its base is moved to a suitable position. For this, the mobile robot makes the base of task robot move to a proper position within the workspace of the task robot. On the other hand, after bringing the task robot within the workspace, we configure TOMM from manipulability ellipsoid determined with respect to the posture of the task robot. And based upon TOMM, the calibration is performed between task robot and mobile robot. This can be described by the following flowchart as shown in Fig. 2 and explained as follows:
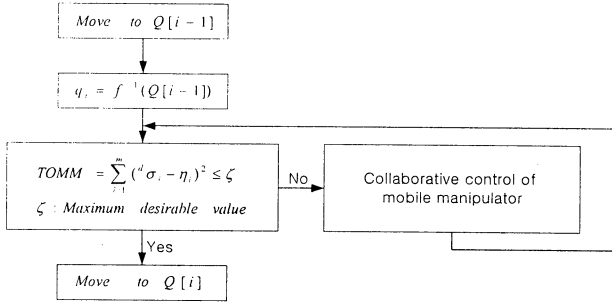

Fig. 2. Flowchart of dynamic cooperation of the mobile manipulator.

1) If a position and the orientation of an end-effector, $Q = [x \ y \ z \ \alpha \ \beta \ \gamma]^T$ is pre-determined, we find the solution ($[q_{t1} \ q_{t2} \ q_{t3} \ q_{t4} \ q_{t5}]^T$) for the task robot. Note that the task free component is required for the existence of solution.

2) We define TOMM from manipulability ellipsoid of the given configuration of the task robot. And when the TOMM is within the desirable range, $\leq \zeta$, the end-effector continuously moves to the next position.

3) In the case of the TOMM is out of the desirable range, that is, the posture of a task robot is not efficient, the dynamic cooperation is performed between the mobile robot and task robot by moving the base of a task robot, i.e., the end-plate of a mobile robot, to take the optimal posture.

4. Simulation

Using the algorithm proposed in the paper, we simulated drawing a big rectangle on the wall.
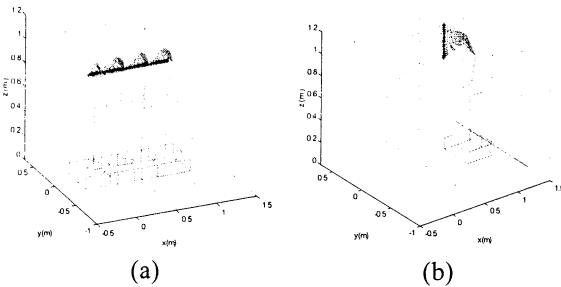

(a)   (b)

Fig. 3. Dynamic cooperation for task segment 1 and 2.

Fig. 3 (a) shows the dynamic cooperation between the mobile robot and task robot when the task robot performs task segment 1 drawing the first horizontal line. When actual TOMM becomes bigger than the max. desirable value in the course of task robot's drawing a horizontal line, the cooperation between the mobile robot and task robot changes the configuration of the task robot suitable to continue the task.

Fig. 3 (b) shows the dynamic cooperation between the mobile robot and task robot when the task robot draws the vertical line (task segment 2).
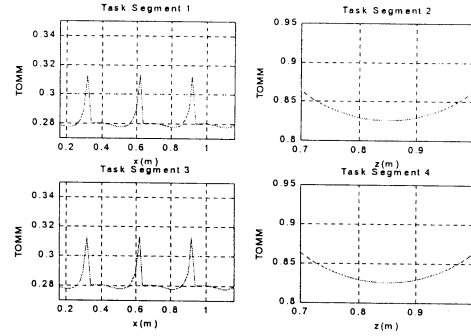
Fig. 4. TOMM for each task segment.

Fig. 4 shows the variation of TOMM for each task segment. In the task segment 1 and 3, there are three peaks that represent the points of cooperation between the task robot and mobile robot. That is, at these points, the configuration of task robot is optimized to continue the task segment through the cooperation. In other words, the actual TOMM is kept below the max. desirable value, $\zeta = 0.3$.

5. Experiment

The mobile robot used in this paper is controlled by low-level controller (87C196CA) included CAN controller and high-level, Pentium 1GHz PC Board. High-level controller determines position/velocity commands and transmits them to low-level controller, and low-level controller performs distributed control for the command of high-level controller. The control of each joint uses the PID control configured for 5msec as the control cycle.

The environment of this experiment is conditioned as follows:

1. Mobile robot moves from the origin of world frame to the initial position to perform the task:

Original position: $(x_i, y_i, z_i) = (0, 0, 0.8)$

Original angle between mobile robot and x-axis: $\pi/8 \ rad$

Initial position: $(x_i, y_i, z_i) = (2, 2, 0.8)$

Initial angle between mobile robot and x-axis: $0 \ rad.$

2. In this experiment, v_x, v_y, and v_z are considered as task constrained components; ω_x, ω_y, and ω_z are considered as task free components; the length of

link and the range of joint variable are set as follows:
d1=0.3, 11=0.2, 12=0.33,

$-80° < q_{t1} < 80°, -35° < q_{t2} < 65°, -100° < q_{t3} < 0°$.

3. The given task is composed of four task segments which are defined as the basic units of task with the same task requirements:

Task segment 1: (2.16, 0.48, 0.7) → (3.16, 0.48, 0.7)
Task segment 2: (3.16, 0.48, 0.7) → (3.16, 0.48, 1.0)
Task segment 3: (3.16, 0.48, 1.0) → (2.16, 0.48, 1.0)
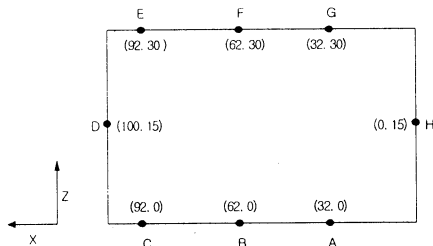Task segment 4: (2.16, 0.48, 1.0) → (2.16, 0.48, 0.7).

Fig. 5. Cooperating points on the X-Z plane.

Table 1. Moving distance and error at the cooperating points.

	A	B	C	D	E	F	G	H
X-axis Coordinate	15.4	16.2	15.5	8.6	-9.0	-15.2	-16.5	-9.2
Z-axis Coordinate	0	0	0	16.1	15.8	0	0	-16.6
X-axis Error	0.4	1.2	0.5	0.6	1.0	0.2	1.5	1.2
Z-axis Error	0	0	0	1.1	0.8	0	0	1.6

Unit (cm)

To draw the rectangle efficiently, the task robot needs to take a suitable initial posture. And then with the TOMM and gradient algorithm described previously, the mobile robot and task robot are moving into their optimal configuration under their collaborative control mode. The cooperating points are represented in Fig. 5, which are determined by the TOMM.

The moving distance and position error to the cooperating points are shown in Table 1.
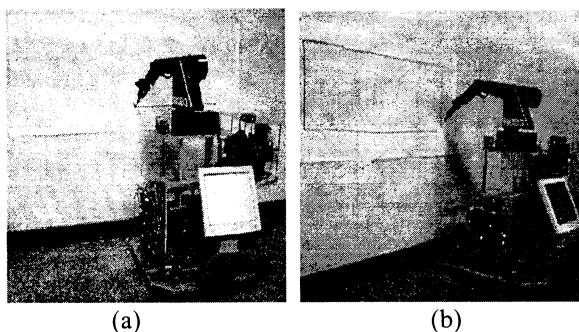

(a)   (b)

Fig. 6. Cooperative drawing by the mobile manipulator.

Fig. 6 shows the task robot takes the optimal configuration for each task segment by the concurrent-cooperation with the mobile robot. As the result, the

mobile manipulator executes the rectangular drawing operation smoothly, even though there exists high positional error at the cooperation points caused by the motion of mobile robot.

6. Conclusions

The effective control methodology of the two serially connected robots – a mobile manipulator – is proposed in this paper. A task can be divided into several task segments according to the required motion components. For each task segment, a desired configuration for the task robot is specified considering the task execution efficiency and the actual configuration is controlled to be close to the desired one. To verify the idea experimentally, a mobile manipulator is implemented by the serial connection of a task robot and a mobile robot that are controlled by a host PC governing the cooperation between them. A host PC can decide the timing to collaborate for the optimal configuration of the task robot. Note that during this cooperation, the end-effector of the task robot is stationary. Even though the mobile robot was suffering from large positional error that came from the slippery of wheels and from the roughness of surface, the conventional PID control algorithm showed reasonable tracking performance. In this paper, even though it was not precise enough, it operated properly to show the idea and to verify the effectiveness of TOMM. As the future research, the dynamic control of the mobile manipulator is left assuming that the precise control of the mobile robot is possible by adding absolute position sensors, for examples, gyro sensors, electrical compass and a CCD camera.

References

[1] Francois G. Pin, "Using Minmax Approaches to Plan Optimal Task Commutation Configuration for Combined Mobile Platform-Manipulator System, "IEEE Trans. Robotics and Automation, vol. 10, no. 1, pp. 44-53, 1994.

[2] Tsuneo Yoshikawa, "Manipulability of Robotic Mechanisms," The International Journal of Robotics Research, vol. 4, no. 2, pp. 3-9, 1985.

[3] Stephen L. Chiu, "Task Compatibility of Manipulator Postures," The International Journal of Robotics Research, vol. 7, no. 5, pp. 13-21, 1998.

[4] M. Aicardi, "Closed-Loop Steering of Unicycle-like Vehicles via Lyapunov Techniques," IEEE Robotics and Automation Magazine, vol. 10, no. 1, pp. 27-35, 1995.

[5] Sukhan Lee and Jang M. Lee, "Task-Oriented Dual-Arm Manipulability and Its Application to Configuration Optimization," Proc. 27th IEEE Int. Conf. on Decision and Control, Austin, TX, Dec. 1988

The 100G Capturing Robot – Too Fast To See –

Makoto KANEKO[†], Mitsuru HIGASHIMORI[†], Reika TAKENAKA[†],
Akio NAMIKI[‡], and Masatoshi ISHIKAWA[‡]

[†]Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima, JAPAN
[‡]Graduate School of Engineering, The University of Tokyo, Bunkyo-ku, Tokyo, JAPAN

Abstract

This paper discusses the capturing robot with the maximum acceleration of 100G in design specification. We find the combination of the arm with the mass of 0.1kg and the spring capable of producing the initial compressed force of 100N, so that we can achieve the 100G. In order to reduce the total capturing time, we newly propose the Arm/Gripper Coupling Mechanism (AGCM) where the spring energy initially accumulated in the arm is transferred to the kinetic energy of the arm and continuously to the kinetic energy for closing the gripper at the capturing point without any time lag. The experimental results show the maximum acceleration of 91G and the capturing time of 25ms. Experiments on capturing a dropping ball are also executed with the assistance of the 1ms-vison.

1 Introduction

In order for a robot to take action quickly as shown in Fig.1, all of computer, sensor and actuation systems should react very quickly. The recent advancement of computer technology is gradually releasing us from the issue on the computation time for controlling robots. As for sensing, various sensors with a high response are now available for chasing a moving object, while the resolution in chasing, of course, depends upon the speed of object. The high speed vision system [1], [2] developed by Ishikawa and his group can track a moving object in every 1msec, which is 33 times faster than conventional vision systems. As a sensor of robot, the 1msec-vision has the potential capability for drastically changing the robotic world due to its extremely fast processing speed. For example, suppose that an object is moving with the speed of 10m/s. During one scanning frame, the object moves only 10mm for 1msec, while it moves even 330mm in conventional ones. The tracking capability with short distance enables us to capture a moving object more exactly by a robot, if the robot actuator can respond quick enough. We are now taking part in a project where we explore how the 1msec-vision can change the robotic world with a combination of a high-speed robot. For executing the project, one issue is that conventional robots are not quick enough for capturing a moving object in the air.

Knowing of the limitation of response time in conventional robots, we begin by designing a new robot system, especially with focusing on the actuation sys-

(a) Initial State (b) Approach Phase (c) Grasp Phase
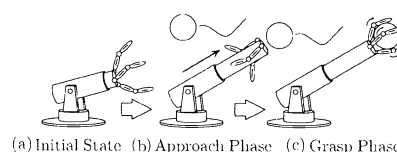
Fig. 1: An Image of Capturing System
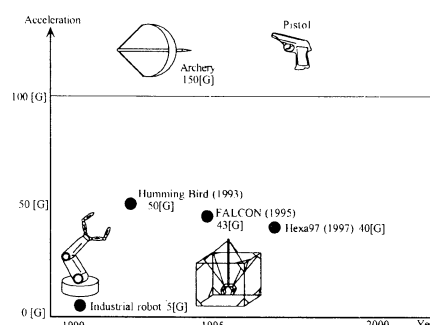
Fig. 2: Map of the Acceleration in Conventional Robot

tem capable of capturing a moving object in the air. For doing it, there are two key issues to be considered: (1) For the arm, how to achieve a quick motion for approaching the target object and (2) for the gripper, how to achieve a quick motion for closing it without any time lag after the arm motion. The accelerations of various robots so far developed are shown in Fig.2, where G denotes the gravitational acceleration. While there have been various robots with extremely high responses, the Humming Bird [3] keeps the world record with the maximum acceleration of 50G. We believe that the maximum acceleration is a good index for evaluating how quickly the robot can respond. In order to drastically increase the maximum acceleration, it is necessary to fully reconsider the actuation system itself, leaving from magnetic based actuators, such as DC or AC servo motor. One good candidate is to utilize an accumulated energy, such as pneumatic or spring energy. As for (1), we can increase the acceleration by using a highly accumulated energy and by decreasing the weight of both arm and gripper. As for (2), we propose a new design where instead of installing actuator individually for both arm

and gripper, the single energy resource is continuously transmitted from the arm to the gripper through a mechanically coupled mechanism. This mechanism is what we call the Arm/Gripper Coupling Mechanism (AGCM) composed of wire, stopper, and transmission plate, where all mechanical components have negligibly small mass. The AGCM is the key for effectively exchanging energy from the arm to the gripper, and as a result, we can remove the time lag between arm and gripper motions.

Based on these ideas, we design and develop the planar type capturing robot composed of DC servomotors for compressing the spring and adjusting the capturing point, a laser sensor for measuring the position of robot arm, an electromagnet for keeping spring energy. The robot can operate in two modes; normal and high-speed modes. The experimental results in high-speed mode show the maximum acceleration of 91G corresponding to almost two times larger than the world record, and the total capturing time of 0.025sec. Experiments on capturing a ball under natural dropping are also demonstrated.

2 Conventional Works

For robot manipulators, there are two mechanical configurations: The serial link manipulator where each link is serially connected from the base to the end-effector through an actuator at each joint, and the parallel link manipulator where the base and the end-effector are connected by link mechanism arranged in parallel. Generally, a parallel link manipulator can react even faster than a serial arm, because we can greatly reduce the mass of moving part by mounting all actuators at the base. We can also classify robot manipulators according to the power transmission mechanism. While there are a number of industrial robots, most of them include the gear train to produce a sufficiently large force at the end point. This is why their accelerations are at most 5G, as shown in Fig.2. A direct drive robot where each motor shaft is directly connected to link can move even quicker than a robot with the gear train. The Barrett Arm [4] with the top speed of 5m/s is categorized into semi-direct drive robots. It is constructed by the combination with powerful motors at the base and tendon drive system without any gear train. The actuator speed is slightly reduced by implementing different size of pulleys between the drive and the output shafts. This design enables us to achieve a high-speed and good controllability with moderate size of actuators.

As for grasping a moving object, Allen and others have demonstrated the automated tracking and grasping of a moving object with a robotic hand [6]. They used a moving model train as an object and therefore, the grasping strategy should not be as difficult as that for capturing an object moving in the air. By using the prototype model of the Barrett Arm [7], Hong and Slotine [8] have succeeded in catching a ball under hand tosses from random locations approximately 1.5-2.5m distance from the base of the arm. As far as we know, this is the first experiment succeeding in catching a moving object in the air by a robot. However, since it takes approximately 0.5sec for completing the whole

catching motion, each toss should be strongly limited in slow speed.

A robot capable of quick action is, of course, desirable from the viewpoint of accurately approaching and catching an object moving fast. In order to realize a quick motion, Kawamura and others [9] have designed an ultra high-speed robot (FALCON) by combining wire transmission with parallel mechanism, and they achieved the maximum acceleration of 45G. Uchiyama and others [10] have succeeded in attaining 40G by utilizing direct drive actuators with parallel mechanism. These maximum accelerations, however, suggest a potential limitation for using electromagnetic based actuators, such as DC or AC motor. On the other hand, an accumulated energy, such as pneumatic or spring one is often utilized especially in legged machines capable of jumping or hopping. Raibert and others [11] first implemented a pneumatic actuator into their one-leg hopping machine and demonstrated fantastic hopping motions. Later, many researchers [12]-[14] followed the same idea for hopping machines

While there are many works concerning with the mechanical design of robot manipulators and hands, our approach is to design both arm and hand in a coupled manner, so that we can remove the time delay between arm and hand motions.

3 Basic Design

We can shoot out an arrow in the maximum acceleration of 150G in archery. To achieve such a high acceleration, there are two keys: The Small Mass of Arrow, greatly contributing to achieving a large acceleration under the same energy resource and the Bow Cord, accumulating a high energy in ready state. Learning such mechanical properties of archery, we design the robot by regarding both arrow and cord as robot arm and spring element, respectively. Additionally, we have to consider the gripper mechanism and the power transmission to it. Especially, from the viewpoint of quick response, we have to design the robot in such a way that we can remove the time lag between the arm and the gripper motions. To cope with this issue, we introduce the Arm/Gripper Coupling Mechanism (AGCM).

3.1 Scale

Let α, m and f be the acceleration of arm including hand, its mass, and the pushing force, respectively. Since $\alpha = f/m$, there are various combinations between m and f for achieving 100G. Letting (m, f) be the combination for achieving 100G, (1Kg, 1000N) or (0.1Kg, 100N) or (0.01Kg, 10N) can be the combination satisfying the requirement. For example, (1Kg, 1000N) means that 100G can be attained when the arm with the mass of 1Kg is pushed by the spring with the initial pushing force of 1000N, though the pushing force of 1000N sounds a bit dangerous when the robot starts unexpectedly. On the other hand, the third combination with the mass of 10g seems difficult to manufacture, since the size of robot should be too small to be realized. Finally, we choose (0.1Kg, 100N) as a moderate combination for achieving 100G.
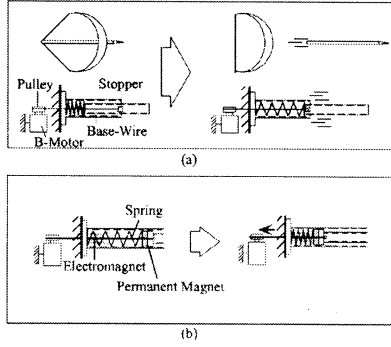
Fig. 3: Robot Mechanism Learnt by Archery

(a) Ready State of the Capturing System

(b) Operation Modes

Fig. 4: AGCM

Fig. 5: Coordinate System of Robot and Object

3.2 Energy Accumulated Arm

Fig.3 (a) shows the relationship between the archery and the basic part of the arm, where a spring is implemented at the base of the robot arm. In archery, the arrow is flew out in the air the moment we release the cord. In order for the robot arm to avoid flying away, the base-wire is connected between the arm and the pulley fixed at the shaft of B-Motor (Base Motor) for adjusting the stop point of the arm. Fig.3 (b) shows more precise illustration of the base, where a permanent magnet and an electromagnet are connected at the end of arm and the base, respectively. Before starting the capturing motion, the B-Motor winds up the base-wire till the spring is fully compressed, as shown in Fig.3 (b). At the same time, the electromagnet is switched on, so that the arm is completely locked. Then, the base-wire is released to the designated length, so that the arm can freely move up to the predetermined point. This is so called ready-phase. The moment the electromagnet is switched into the opposite polarity, the arm quickly starts in maximum acceleration and stops at the position in the predetermined length of the base-wire. Without any additional mechanism, the potential energy accumulated in the ready phase is completely lost and changed into heat, when the arm stops. As a result, we have to prepare another energy resource for operating the gripper, which will cause the increase of mass of moving part as well as the increase of the time lag between the arm and the gripper motions. These are, of course, not desirable from the viewpoint of both increasing the acceleration and decreasing the response time. The AGCM is the key for transmitting the energy from the spring to the arm, and from the arm to the gripper continuously, with a simple mechanism based on wire.

3.3 Arm/Gripper Coupling Mechanism (AGCM)

The gripper should have a small mass and a simple structure, so that it can quickly move. We utilize so-called the Soft Gripper proposed by Hirose [5], since it satisfies our requirements, such as small mass and a simple operation. It can operate by simply pulling and releasing the drive-wire. We implement a rota-
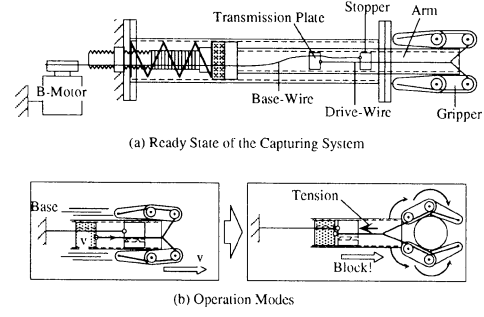
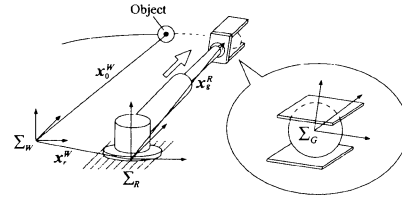tional spring in each joint, so that the returning motion can be produced automatically when the drive-wire is released. Now, a question is how to transmit the energy from the arm to the gripper at the capturing point. This question can be replaced by how to pull the drive-wire at the predetermined point. For this question, we finally come to the simple mechanism as shown in Fig.4 where the base wire is connected to the stopper through the hole of the transmission plate. In the ready state, the base-wire is loosen enough so that the arm can freely move until it is fully blocked by the stopper. The moment the polarity of electromagnet is changed, the arm is accelerated with the transmission plate as well as the gripper. Although the inertia force caused by the transmission plate pulls the drive-wire in such a way that the gripper may be closed, we can choose appropriate rotational springs of gripper, for overcoming the moment coming from the inertia force. When the arm reaches the predetermined length as shown in Fig.4 (b), the transmission plate can no more move due to physically blocking by the stopper. At this moment, a large interaction force suddenly appears between them. As a result, a large tension appears in the drive-wire and makes the gripper close quickly without any time lag. This is what we call the AGCM. By utilizing the AGCM, the kinetic energy of the arm is transmitted to the drive-wire connected to the gripper through the transmission plate and the stopper. We would note that the compression of spring at the end of capturing motion produces the grasping force, while most of initial energy finally changes into the heat.

4 Some Remarks for Capturing a Moving Object

We first define $V(x^G)$ as the scalor function expressing the region enabling the robot to capture an object irrespective of orientation of object, where x^G denotes the vector whose origin is the geometrical center of gripper and $\sum_G$ is the gripper coordinate system. Mathematically, $V(x^G)$ is given by,

$$V(x^G) = \begin{cases} V(x^G) > 0 & : \text{Outside of } V(x^G) \\ V(x^G) = 0 & : \text{Surface of } V(x^G) \\ V(x^G) < 0 & : \text{Inside of } V(x^G) \end{cases} \quad (1)$$

Generally, $V(x^G)$ becomes small as the size of object increases and finally results in zero. Now, imagine that an object is moving in 3D space as shown in Fig.5, where $\sum_W$, $\sum_R$, $x_o^W \in R^3$, $x_r^W \in R^3$, $x_g^R \in R^3$, $R_R^W \in R^3$ and $R_R^G \in R^3$ are the world coordinate system, the robot coordinate system, the position vector of object in $\sum_W$, the position vector indicating the origin of $\sum_R$ in $\sum_W$, the position vector of the center of gripper in $\sum_R$, the rotational matrix from $\sum_W$ to $\sum_R$ and the rotational matrix from $\sum_G$ to $\sum_R$, respectively. The center of gripper is given by

$$x_g^W = x_r^W + R_R^W x_g^R \quad (2)$$

where x_g^W denotes the position vector of the center of gripper in $\sum_W$. We assume that the gripper plate is massless and therefore it can move infinite velocity, while we suppose the inertia of arm is not negligible. As the necessary condition for catching, the gripper has to meet with a moving object. This condition is given as follows.

[Necessary Condition for Catching an Object]
In order for a robot to capture a moving object, there should exist $\exists t$, such that it can satisfy

$$V(\Delta x^G) \leq 0 \quad (3)$$

$$\Delta x^G = (R_G^R)^T (R_R^W)^T \{x_g^W(t) - x_o^W(t)\} \quad (4)$$

where t and T denote time and the transpose of matrix.

Without satisfying this condition, the robot can never meet a moving object within the capturing region $V(x^G)$. However, note that this condition is not sufficient for stopping the object within the gripper. Because due to the inertia force, the object will slide even after the object is in contact with the inner surface of gripper. The sliding distance depends on the contact friction, the friction coefficient, and the velocity of object. Since we can not estimate the exact sliding distance in advance, the high speed capturing system should be designed so that the gripper may have an appropriate area for the gripper plate.
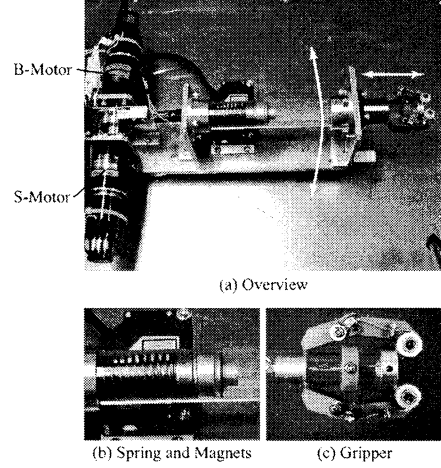

(a) Overview

(b) Spring and Magnets   (c) Gripper

Fig. 6: Experimental System
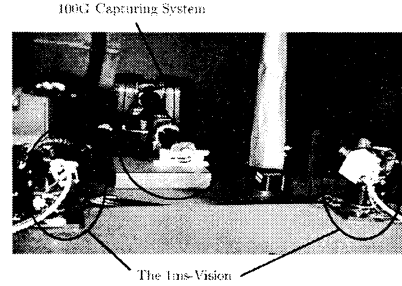
Fig. 7: Experimental Set Up for Capturing a Dropping Ball

5 Experiments

5.1 Experimental System

Fig.6 shows the experimental system, where (a), (b) and (c) are an overview of the system, the spring and the magnets, and the gripper, respectively. There are three DC servo-motors: the S-Motor for compressing the spring, the B-Motor for adjusting the capturing point, and the R-Motor for rotating the whole system. The robot can operate in two modes, the normal mode where we can use it as a normal planar robot by independently controlling both S- and B-Motors, and the high-speed mode where the S-Motor is released enough so that the stopping point can be determined by the B-Motor only. The spring constant, the mass for both arm and grippers, and the absorption force between the electromagnet and the permanent magnet are 0.65N/mm, 100g, and roughly 71N, respectively. These specifications tell us the maximum acceleration is 73G under neglecting the effect on acceleration due to magnetic force. For detecting the position of the target object, we implement the 1ms-vision whose sampling time is 1ms. A laser sensor (KEYENCE LB-1000) is also installed at the base lo-

Fig. 8: $V'(x^G)$

Fig. 9: Position and Velocity of Arm during a High-speed Mode

Fig. 10: Simulation
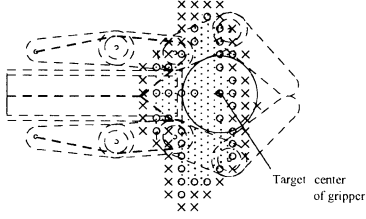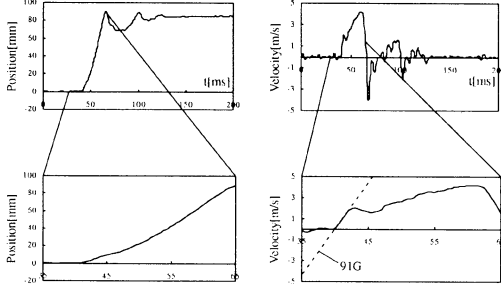

Fig. 11: The Relationship between the Ball Position and the Captured Image

cated just below the arm so that we can measure the position of robot arm with a reflector. Fig.7 shows the overview of the experimental set up for capturing a dropping ball where the only capturing part is implemented to the end of the Barrett Arm[4] and two 1ms-visions are implemented so that we can obtain the 3D position of object.

5.2 Experimental Results

We first choose the cylindrical object with the diameter of 30mm. We define the 2D capturing region $V'(x^G)$ where the target center of gripper denotes the reaching position of the center of gripper under no object. The robot can capture successfully the object when it is perpendicularly placed on the table and the center exists within $V'(x^G)$. Fig.8 shows the experimental results where the robot can capture the object when it is placed on the circle(○) points and the robot fails in capturing it when it is placed on the cross(×) points. Fig.9 shows both position and velocity during the capturing motion for the cylindrical object, where the position and the velocity data are obtained by the laser sensor and the numerical differentiation of the position signal, respectively. This figure tells us that the time taken for the capturing motion is roughly 0.03sec. We can observe the maximum acceleration of 91G in the initial phase and the average acceleration of roughly 22G. An interesting observation is that the maximum acceleration observed by the experiment is 25% more than that predicted by an ideal model. This difference comes from that in the ideal model we neglect the effect on acceleration due to magnetic force immediately after changing the polarity. To validate this consideration, we show a simulation result in Fig.10 where the magnetic force is neglected. By
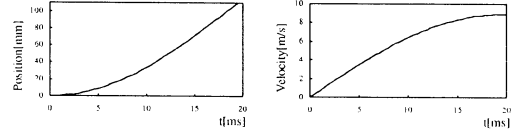
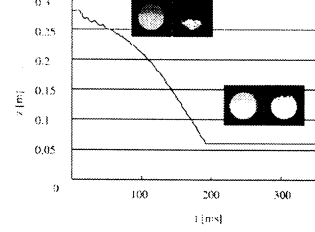comparing both Fig.9 and Fig.10, the effect on acceleration by magnetic force is clearly observed in the experiment especially at the initial phase, while it does not in the simulation.

By using the robot, we also challenge a difficult task where the robot catches a ball dropping from an appropriate height. The 1ms-vision mounted at the base continuously provides us with the center of gravity of object captured by camera. Fig.11 shows the trajectory obtained by the 1ms-vision where both the real and the 2-bits visual images are shown. From Fig.11, we would note that the 2-bits image is heavily influenced by the illumination condition. For example, it can represent only the bottom part of object when the ball is dropping at the height of $z=0.25$m, while it shows the original ball correctly when it is at $z=0.10$m. This means that whether the position of ball can be measured appropriately or not strongly depends on the condition of illumination. This is the reason why the detected position of ball does not change smoothly at the beginning of dropping motion. Through the observation of high speed camera, we learn that, it takes 25ms for completing the capturing motion after we send the starting command to the robot. Therefore, we have to estimate when the ball reaches the capturing region of robot by utilizing the ball positions so far obtained. We utilize the least-square-method for estimating the time. Fig.12 shows the positions of ball obtained by camera and the estimated time to the capturing region, where z, T_c and T_c are the ball position, the estimated time to the capturing region and the actual captured time, respectively. All ball positions are utilized for estimating T_c in Fig.12(a), while the ball positions whose 2-bits image has more than 50 pixels are utilized for computing the time to the capturing point in Fig.12(b). Since we can remove the ball positions whose illumination conditions are not good enough, the estimated time to the capturing point should be more reliable in Fig.12(b) than that in Fig.12(a). As indicated in Fig.12, the difference be-
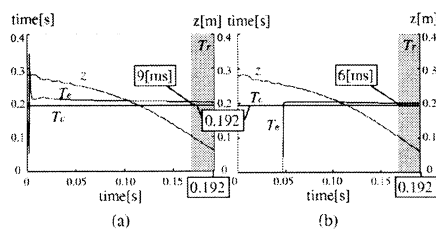
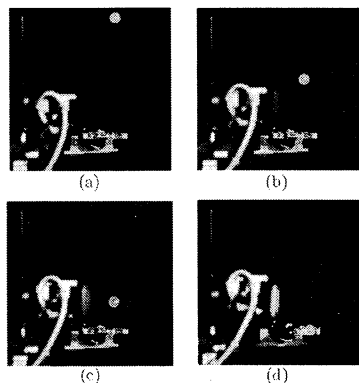**Fig. 12**: The Estimated Time with Respect to Time

Fig. 13: A Series of Pictures for Capturing a Dropping Ball

tween the estimated time and the actual time is 9msec in Fig.12(a), while it is 6msec in Fig.12(b). Now, suppose that we naturally drop the ball at the height of 1.0m. When the ball passes the capturing point, it has the speed of roughly 4.4m/sec, namely 4.4mm/msec. As the vertical width of the gripper is 15mm, the ball passes through the width during 3msec. Actually, it may be difficult to keep such a high accuracy for estimating the time to the capturing point. Considering the repeatability of the shift between the estimated time T_e and actual time T_c, we regard the mean time shift between them as a sort of calibration data. Fig.13 shows a series of pictures of ball capturing experiment where the robot can successfully catches the ball with the assistance of 1ms-vision. Currently, the success rate for capturing the object is roughly 50%. We believe that the success rate will be greatly improved by more accurate calibration and by high motion repeatability of robot. Since the capturing motions are too fast, we can not see what is really happening. In order clearly to observe the capturing motions, we took the continuous motions by the high speed camera whose frame rate is 1000 frames/sec.

6 Conclusion

We discussed the design of high-speed capturing robot by using single energy resource. We proposed the Arm/Gripper Coupling Mechanism (AGCM), so that we can transmit the initially accumulated energy from the arm to the gripper. The experimental results

by the developed robot showed the maximum acceleration of 91G corresponding to almost two times higher than the conventional world record, and the total capturing time of 0.025sec. Experiments on capturing a ball were also shown.

References

[1] A. Namiki, Y. Nakabo, I. Ishii, and M. Ishikawa. 1-ms Sensory-Motor Fusion System. *IEEE Trans.on Mechatronics, vol.5, no.3, pp244-253*, 2000.

[2] I. Ishii, and M. Ishikawa. Self Windowing for High Speed Vision. *in Proc. of the IEEE Int. Conf. on Robotics and Automation, pp1916-1921*, 1999.

[3] J. P. Karidis, G. McVicker, J. P. Pawletko, L. C. Zai, M. Goldowsky, R. E. Brown, R. R. Comulada. The Hummingbird Minipositioner Providing Three-Axis Motion At 50 G's With Low Reactions. *IEEE Int.Conf.on R&A, pp685-692*, 1992.

[4] http://www.barretttechnology.com/ robot/products/arm/armspfr.htm.

[5] S. Hirose and Y. Umetani. Soft Gripper. *Proc. of ISIR, pp112-127*, 1983.

[6] P. K. Allen, A. Timocenko, B. Yoshimi, and P. Michelman. Automated Tracking and Grasping of a Moving Object with a Robotic Hand-Eye System. *IEEE Trans.on R&A, vol.9, no.2, pp152-165*, 1993.

[7] W. T. Townsend. The Effect of Transmission Design on Force-Controlled Manipulator Performance. *PhD Thesis, Department of Mechanical Engineering, MIT*, 1988.

[8] W. Hong, J. E. Slotine. Experiments in Hand-Eye Coordination Using Active Vision. *Lecture Notes in Control and Information Sciences, vol.223, pp130-139*, 1997.

[9] S. Kawamura, W. Choe, S. Tanaka, and S. R. Pandian. Development of an Ultrahigh Speed Robot FALCON Using Wire Drive System. *in Proc. of the IEEE Int. Conf. on Robotics and Automation, pp215-220*, 1995.

[10] M. Uchiyama, T. Sadotomo, and K. Masukawa. Dynamic Control Experiment on a Parallel robot HEXA. *J. of Robotics Society of Japan, vol.14, no.2, pp297-304*, 1996.

[11] M. H. Raibert. Dynamically Stability and Resonance in a Legged Hopping Machine. *IFToMM Proc. of Symp. on Theory and Practice of Robots and Manipulators, pp352-367*, 1983.

[12] M. H. Raibert. Trotting, Pacing, and Bounding by a Quadruped Robot. *J. of Biomechanics, 23, pp79-98*, 1990.

[13] M. D. Berkemeier and K. V. Desai. Design of a Robot Leg with Elastic Energy Storage. *in Proc. of the IEEE Int. Conf. on Robotics and Automation, pp213-218*, 1996.

[14] B. V. Chapnik, G. R. Heppler, and J. D. Aplevich. Modeling Impact on a One Link Flexible Robotic Arm. *IEEE Trans.on R&A, vol.7, no.4, pp479-488*, 1991.

Effectiveness of Integration of Skill Techniques in Manipulation Robots

°Akira NAKAMURA, Kosei KITAGAKI and Takashi SUEHIRO

National Institute of Advanced Industrial Science and Technology (AIST)
AIST Tsukuba Central 2, 1-1-1 Umezono, Tsukuba, Ibaraki, 305-8568 Japan
E-mail: a-nakamura@aist.go.jp, k.kitagaki@aist.go.jp, t.suehiro@aist.go.jp

Abstract

In recent years, robots have been rapidly introduces into several fields. In order to play a part in extensive fields, manipulation robots need to perform various tasks using special techniques. We call these "skills." Skills can yield easier operation and proper sequence to accomplish the task. Therefore, the reliability of task achievement can be increased by using skills. We will now consider that manipulation task is carried out by four steps of sensing, modeling, planning and execution. We have proposed several skill techniques for these modules composed of manipulation task. In this paper, we describe high reliability of task achievement obtained by integrating the skill techniques in these modules.

Key words: skill, manipulation, geometric model, planning, visual sensing

1. Introduction

Dexterous manipulation plays an important role in working robots. For manipulating robots to be useful in several fields, it is necessary to realize various tasks by using special techniques. We called these "skills." Manipulation can be simplified and high reliability of task achievement can be obtained by using skills. We assume that manipulation task can be carried out by four steps of visual sensing, geometric modeling, planning and execution [1], [2]. We have proposed several skill techniques for these modules composed of manipulation task.

Especially, we have shown many skills in execution of manipulation [3]-[5]. By analyzing human motions during tasks such as assembly and disassembly, movements were found to consist of several significant motion primitives. We called these "skills" in the execution step and showed that most of the tasks of the manipulator can be composed of sequences of skills. That is, we demonstrated that robots can perform various human tasks by using the concept of skill.

We have proposed skill techniques in each module such as modeling, planning and execution and considered that each skill has an effect on the respective modules. In this paper, we describe high reliability of task achievement obtained by integrating the skill techniques in these modules.

The next section explains exact process of sensing, modeling, planning and execution. Skill in each module is shown in section 3, 4, 5 and 6, respectively. In section 7, effectiveness of integration of skills and conclusion are shown.

2. Process of Sensing, Modeling, Planning and Execution

In our method, manipulation task is composed of visual sensing, geometric modeling, planning and execution. We will now describe the procedure of sensing, modeling, planning and execution for a manipulation task in detail. Figure 1 shows the minute procedures [1], [2]. In this scheme, planning of the task level is first performed, and then executions of skill level are carried out according to sequences derived from the task planning.
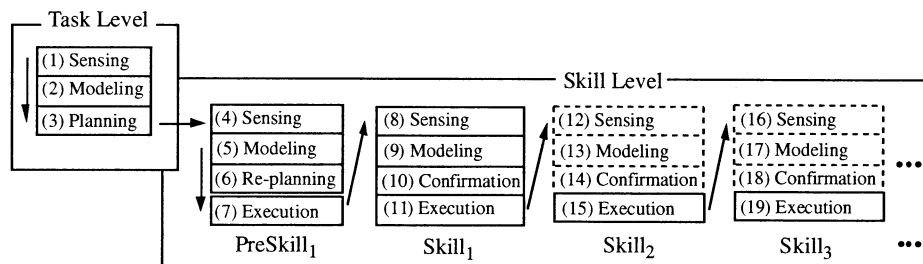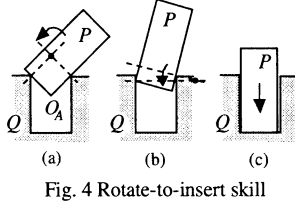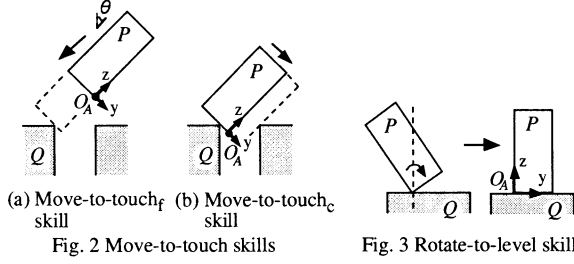

Fig. 1 Process flow

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

Fig. 2 Move-to-touch skills   Fig. 3 Rotate-to-level skill

(a)   (b)   (c)

Fig. 4 Rotate-to-insert skill

3. Skills in Execution

This section explains our concept of skills in execution of manipulation. See References [3]-[5] for more details.

3.1. Skill primitives

In assembly and disassembly tasks, skills in which the contact states vary are particularly significant. In References [3]-[5], we considered three skills, "move-to-touch," "rotate-to-level" and "rotate-to-insert," which play an important part in such tasks. For simplicity, we assume that a grasped object, another object and the hole are rectangular parallelepipeds.

(1) Move-to-touch Skill

The move-to-touch skill is defined as the transition of a grasped object P in a constant direction that continues until a contact with another object Q occurs (Fig. 2(a)). This skill is performed in velocity control mode. Not only the transition in free space (Fig. 2(a)), but also the sliding while keeping contact in a different direction of motion (Fig. 2(b)) is included in this skill. These two transitions are represented respectively by the move-to-touch$_f$ skill and the move-to-touch$_c$ skill. The achievement of this skill can be detected by instantaneously increased resistance in the direction of the transition.

(2) Rotate-to-level Skill

The rotate-to-level skill is defined as rotation motion around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q (Fig. 3). This skill is performed with a pushing force. The achievement of this skill can be detected by the change of the instantaneous center position.

(3) Rotate-to-insert Skill

In an insertion task it is generally difficult to
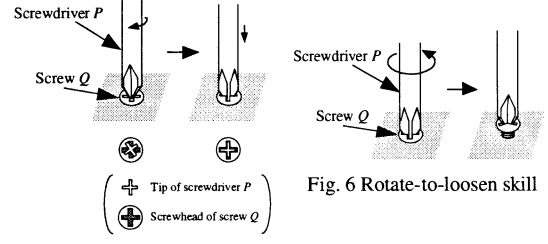
achieve the state in Fig. 4(b) directly. The state in Fig. 4(a) is achieved first by using the skill sequence of Fig. 2(a) and (b). The state in Fig. 4(b) is then accomplished by gradually raising the object while maintaining contact as in Fig. 4(a). The rotate-to-insert skill is this motion of rotating the object P obliquely into the hole in another object Q to insert it accurately. In our study, we assume that the rotate-to-insert skill also includes the pressing motion required to achieve the goal of the insertion task (Fig. 4(c)). The achievement of each rotate-to-insert skill is detected by the variation of the instantaneous center position.

Furthermore, we will consider two skills related to the tasks in the examples mentioned later. In these two skills, we use a cross-head screwdriver and screw according to tasks as examples.

(4) Rotate-to-bite Skill

This skill is a rotation motion around the axis of the screwdriver to fit the tip of the screwdriver into the flutes of the screw head (Fig. 5). This skill is performed with pushing force.

(5) Rotate-to-loosen Skill

This skill is defined as a rotation motion to loosen the screw (Fig. 6). This is performed by matching the axes of rotation of a part and a tool. If these axes do not correspond, the tool is moved to the position before execution of rotation. In this paper, we assume that this skill also includes the transition to remove an error before rotating.

3.2. Composition of skill sequence

A specific task is composed of sequences of skill primitives such as move-to-touch, rotate-to-level and rotate-to-insert skills. The skill sequences can be decided by several methods. We showed a method using variations of the number of contact points in skill primitives [2].

4. Skills in Planning

In this section, we will discuss the trajectory of skill motions in the configuration space and planning using backprojection.
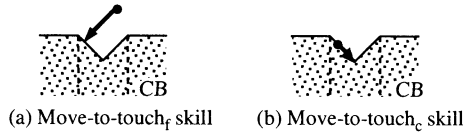

Fig. 5 Rotate-to-bite skill

Fig. 6 Rotate-to-loosen skill

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

Fig. 7 Move-to-touch skills in C-space

Fig. 8 Move-to-touch$_f$ skill with control uncertainty

(1) Move-to-touch Skill

In the configuration space the trajectories of the object P being manipulated with the move-to-touch$_f$ skill in Fig. 2(a) and the move-to-touch$_c$ skill in Fig. 2(b) are drawn in Fig. 7(a) and (b), respectively. CB is a C-obstacle which represents an object Q in C-space based on the reference point O_A of object P. To take into account the uncertainty of control when using the move-to-touch$_f$ skill, we have drawn the trajectory using the control uncertainty cone shown in Fig. 8.

(2) Rotate-to-level Skill

Assuming that reference point O_A is a vertex in prior contact with the surface (Fig. 3), the position of O_A on the YZ-plane in the configuration space stays constant (Fig. 9).

(3) Rotate-to-insert Skill

The trajectory of the object in the configuration space when manipulated with the rotate-to-insert skill is shown in Fig. 10. The transfer motion of the vertex from Fig. 4(a) to Fig. 4(b) happens at orientation $\theta = \theta_t$ in Fig. 10, where the phase of C-obstacle CB changes.

The backprojection of each skill is derived by the reverse trajectory from each goal [6]. For example, the backprojection of the move-to-touch$_f$ skill (Fig. 11(a)) is drawn by projecting a similar velocity cone from the goal edge (Fig. 11(b)).

Next, a command sequence to perform a specific task is created using skill primitives expressed by trajectories in the configuration space. The method to decide the most suitable command sequence and most suitable initial state is shown in [6]. The most suitable sequence by which the largest region of backprojection can be obtained and the instructed task can actually be performed is chosen. The most suitable initial state is chosen from the viewpoint of task achievement reliability as shown in [6].

Our method can be considered as skillful planning, since the most suitable command sequence and the most suitable initial state are derived easily.
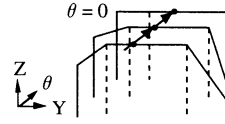

Fig. 9 Rotate-to-level skill in C-space
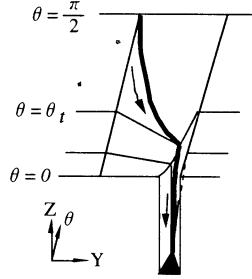
Fig. 10 Rotate-to-insert skill in C-space

(a) Move-to-touch$_f$ skill   (b) Backprojection
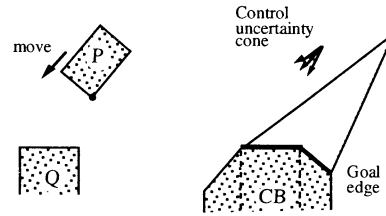
Fig. 11 Backprojection of move-to-touch$_f$ skill with uncertainty of control

5. Skills in Geometric Modeling

In planning for manipulation and visual sensing for model matching, it is not necessary to use geometric models which express completely real objects. By constructing geometric models using only the data necessary for planning and visual sensing, it is possible to facilitate these processes. We call such robust models "simplified geometric models (SGM)." In this paper, skills in geometric modeling means facilitation of planning and visual sensing using simplified geometric models. Simplified geometric models in these two types are defined as follows.

Simplified geometric models in planning (SGM_p) are geometric models composed of the necessary and minimum data for shape, position and orientation to perform skill-based planning. Simplified geometric models in visual sensing (SGM_v) are geometric models composed of the necessary and minimum data for shape, position and orientation to perform model matching. Simplified geometric models in planning (SGM_p) are constructed from the data of simplified geometric models in visual sensing (SGM_v).

We will consider the task of loosening a screw using a screwdriver for examples [2]. Although there are many kinds of screws and screwdrivers, we will consider this task by unifying skill sequences in the following examples.

Table 1 Simplified geometric models in the task of loosening a screw using a screwdriver

	RO	Skill₁ (Preskill₁)		Skill₂		Skill₃	
		SGM$_p$	SGM$_v$	SGM$_p$	SGM$_v$	SGM$_p$	SGM$_v$
(a) Phillips				Not needed	Not needed	Not needed	Not needed
(b) Reed and Prince				Not needed	Not needed	Not needed	Not needed
(c) Slotted				Not needed	Not needed		
(d) Robertson				Not needed	Not needed	Not needed	Not needed

RO : Real objects
SGM$_p$: Simplified geometric models for planning
SGM$_v$: Simplified geometric models for visual sensing and model matching

Fig. 12 Backprojection in the case that the directions of the visual sensing and manipulation are the same

We assume that the task of loosening a screw using a screwdriver is composed of the following skills.

Skill₁ : Move-to-touch skill
Skill₂ : Rotate-to-bite skill
Skill₃ : Rotate-to-loosen skill

In this case, Table 1 shows simplified geometric models of screws used in modeling in each skill [2].

6. Skills in Visual Sensing

It is important to take into account position and orientation of visual sensing for high reliability of task achievements. It needs to be stiff for sensing errors. In this paper, skills in visual sensing mean sensing techniques of 3D data to increase the reliability.

We will consider backprojection that takes into account visual sensing errors. In general, position data obtained by a vision system will have large errors in depth data. First, for simplicity, we consider only the sensing errors in depth from a vision system. When the directions of the visual sensing and manipulation are the same, the size of the backprojection is largest (Fig. 12) [1]. However, it is difficult to achieve visual sensing from this viewpoint because of collision avoidance between the vision system and manipulator. Therefore, visual sensing should be performed as closely as possible from the direction of manipulation. While we considered errors in the depth direction, if it is necessary to take into account errors in other directions, backprojection can be similarly derived by changing the uncertainty of a point to an elliptical region.

7. Effectiveness in Integration of Skill Techniques and Conclusion

We have considered skill in each module in sections 3, 4, 5 and 6; i) Manipulators are capable of performing various human tasks using sequences of skills in execution. ii) The most suitable sequence and the most suitable initial state are derived easily using skills in planning. iii) Visual sensing and planning are facilitated using skills in modeling. iv) Skills in visual sensing yield stiffness for sensing errors. These skills can increase reliability of task achievement.
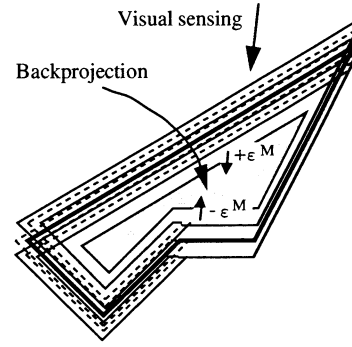
Skill techniques in sensing, modeling and planning in sections 4, 5 and 6 are based on the concept of skills in execution of manipulation. Therefore, these skill techniques are restricted within narrow limits. In the future, we will study higher reliability of task achievement obtained by integrating organically the skills in four modules, development of many skill primitives in the modules, and application for real manipulation systems.

References

[1] Nakamura A, Suehiro T, Machida K (2001), Fine-motion planning using skill-based backprojection with uncertainties in control, visual sensing and model. In: Bejczy A. K, Kozlowski K, Rudas I. J (eds) Proceedings of the 10th International Conference on Advanced Robotics (ICAR 2001), Budapest, Hungary, Aug. 22-25, 2001, pp 541-548

[2] Nakamura A, Ogasawara T, Kitagaki K, Suehiro T (2001), Using robust and simplified geometric models in skill-based manipulation. In: Tarn T. J, Burdick J, Papanikolopoulos N (eds) Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), Hawaii, USA, Oct. 29-Nov. 3, 2001, vol. 1, pp 138-145

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

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

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

[6] Nakamura A, Ogasawara T, Suehiro T, Tsukune H (1996), Skill-based backprojection for fine motion planning. In: Asada M, Arai T, Kak A, Sandini G (eds) Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '96), Osaka, Japan, Nov. 4-8, 1996, pp 526-533

Robot Assisted Activity at a Health Service Facility for the Aged

Kazuyoshi Wada[*1,2], Takanori Shibata[*1,3] Tomoko Saito[*1], Kazuo Tanie[*1,2]

*1 Intelligent Systems Institute, AIST
1-1-1 Umezono, Tsukuba, Ibaraki, 305-8568 Japan
*2 Institute of Engineering Mechanics, University of Tsukuba
*3 PRESTO, JST

{k-wada, shibata-takanori, tomo-saito, tanie.k}@aist.go.jp

Abstract

We have been developing mental commit robots that provide psychological, physiological, and social effects to human beings through physical interaction. The appearances of these robots look like real animals such as cat and seal. The seal robot was developed especially for therapy. We have applied seal robots to assisting activity of elderly people at a health service facility for the aged. In order to investigate psychological and social effects of seal robots to the elderly people, we evaluated elderly people's moods by face scales (which express person's moods by illustration of person's faces). Seal robots were provided into the facility for three weeks. As the results, feelings of elderly people were improved by interaction with the seal robots.

Fig.1 Seal Robot "Paro"

1. Introduction

Recently, animal assisted therapy and activity are becoming popular at hospital and nursing home, especially in the United States [1]. A doctor or nurse makes a program for therapy. Following three effects are expected in animal assisted therapy and activity:

(1) Psychological effect (e.g. relaxation, motivation)
(2) Physiological effect (e.g. improvement of vital sign)
(3) Social effect (e.g. activation of communication among inpatients and caregivers)

However, most hospitals and nursing homes, especially in Japan, don't accept animals even though they admit effects of animal assisted therapy and activity. They are afraid of negative effects of animals to human beings such as allergy, infection, bite, and scratch.

We have been building animal type robots as examples of artificial emotional creatures [2-5]. The animal type robots have physical bodies and behave actively while generating goals and motivations by themselves. They interact with human beings physically. When we engage physically with an animal type robot, it stimulates our affection. Then we have positive emotions such as happiness and love, or negative emotions such as anger
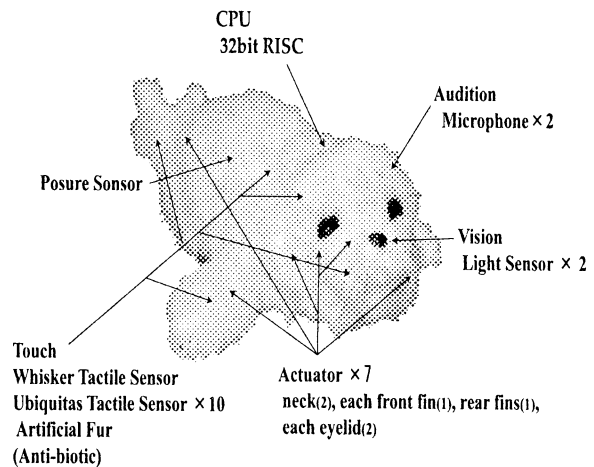
and fear. Through physical interaction, we develop attachment to the animal type robot while evaluating it as intelligent or stupid by our subjective measures. In this research, animal type robots that give mental value to human beings are referred to as "mental commit robot." We have developed cat robot and seal robot as the mental commit robot.

We have applied seal robots as substitution of real animals to therapy of children at a university hospital [3]. This was referred to as robot-assisted therapy (RAT). Moods of children were improved by interaction with the robot. Moreover, the robot encouraged children to communicate with each other and caregivers.

In addition, we have applied seal robots to robot-assisted activity (RAA) for elderly people [4, 5]. The robots improved their moods and brought vigor to them. Moreover, nursing staff's mental poverty decreased because the elderly people spent their time by themselves with the robots.

In this paper, we applied seal robots to assist activity of elderly people at a health service facility for the aged, in order to investigate psychological and social effects of

seal robots to the elderly people. Then, we compared with effects of the seal robot and those of a placebo seal robot that was changed its motion generation program.

Chapter 2 explains a seal robot and placebo seal robot that were used for RAA. Chapter 3 describes ways of experiments and explains the effects of RAA to elderly people. Chapter 4 discusses current results of RAA and future works. Finally, chapter 5 concludes this paper.

2. Seal Robot and Placebo Seal Robot

2.1. Specifications of Seal Robot

Seal robot, Paro was developed to have physical interaction with human beings (Fig.1). Paro's appearance is from a baby of harp seal, which has white fur for three weeks from its born. As for perception, Paro has tactile, vision, audition, and posture sensors beneath its soft white artificial fur. In order for Paro to consist of a soft body, a tactile sensor was developed and implemented. As for action, it has seven actuators; two for each eyelids, two for neck, one for each front fin, and one for two rear fins. Weight of Paro is about 3.0 [kg].

Paro has a behavior generation system that consists of hierarchical two layers of processes: proactive and reactive processes. These two layers generate three kinds of behaviors; proactive, reactive, and physiological behaviors:

(1) Proactive Behaviors: Paro has two layers to generate its proactive behaviors: behavior-planning layer and behavior-generation layer. Considering internal states, stimuli, desires, and a rhythm, Paro generates proactive behaviors.

(a) Behavior-planning layer: This has a state transition network based on internal states of Paro and Paro's desire produced by its internal rhythm. Paro has internal states that can be named with words of emotions. Each state has numerical level and is changed by stimulation. The state decays by time. Interaction changes internal states and creates character of Paro. The behavior-planning layer sends basic behavioral patterns to behavior-generation layer. The basic behavioral patterns include some poses and some motions. Here, although "proactive" is referred, proactive behaviors are very primitive compared with those of human beings. We implemented similar behaviors of a real seal into Paro.

(b) Behavior generation layer: This layer generates control references for each actuator to perform the determined behavior. The control reference depends on strength of internal states and their variation. For example, parameters change speed of movement, and the number of the same behavior. Therefore, although the number of basic patterns is countable, the number of emerging behaviors is uncountable because numeral parameters are various. This creates living like behaviors. In addition, as

for attention, the behavior-generation layer adjusts parameters of priority of reactive behaviors and proactive behaviors based on strength of internal states. This function contributes to situated behavior of Paro, and makes it difficult for a subject to predict Paro's action.

(c) Long-term memory: Paro has a function of reinforcement learning. It has positive value on preferable stimulation such as stroked. It also has negative value on undesirable stimulation such as beaten. Paro put values on relationship between stimulation and behaviors. Gradually, Paro can be shaped to preferable behaviors of its owner.

(2) Reactive behaviors: Paro reacts to sudden stimulation. For example, when it hears big sound suddenly, Paro pays attention to it and looks at the direction. There are some patterns of combination of stimulation and reaction. These patterns are assumed as conditioned and unconscious behaviors.

(3) Physiological behaviors: Paro has a rhythm of a day. It has some spontaneous desires such as sleep based on the rhythm.

2.2. Specifications of Placebo Seal Robot

We often experience that we lose interest in toys when we found its mechanism. Therefore, we consider following hypothesis:

The robots that execute only defined simple motions are predicted its motions by people, and they lose interest in the robots. Moreover, the robots also lose its effects to the people.

According to this hypothesis, we changed regular Paro's program, and made placebo Paro as follows.

Proactive behaviors: repetition of following five kinds of actions.

(1) Blink
(2) Swing rear fin to right and left
(3) Swing both front fins to forward and backward
(4) Swing head to right and left
(5) Cry

Reactive behaviors: following simple reactions against stimuli.

(1) Cry (sound is different from proactive motion's cry)
(2) Raise head

3. Robot Assisted Activity for Elderly People

We applied Paro to robot-assisted activity for elderly people at a health service facility for aged in order to investigate its effects on elderly people. The health service facility for aged is an institution that provides several services, such as stay in the institution, day care and rehabilitation to elderly people. People who need nursing can stay in there during a certain period. In order to rehabilitate into society, they are provided daily care and trained to be able to spend their daily life

independently during their staying at the institution. When we started experiment at the institution, about 100 elderly people were staying in there. Moreover, about 30 people of them were dementia. People who were not dementias stayed in A and B building. On the other hand, people who were dementia stayed in C building, and they were isolated from other people.

Before starting the robot-assisted activity, we explained the purposes and ways of the experiment to elderly people who stayed A and B building, and received their approval. Symptoms of the elderly people who approved the investigation were various with different reasons (no answer to questionnaires, bedridden, etc). Some people were impossible to be investigated. Then, a nursing staff that knew usual states of the elderly people well evaluated them, and decided who could be investigated. After the evaluation, the number of subjects was 23. 12 subjects stayed in A building, and 11 subjects were in B building. Their basic attributes are shown in Table 1.

3.1. Ways of activity

Regular Paro was provided to the subjects who stayed in B building, and placebo Paro was provided to the subjects who stayed in A building. In order to prevent that subjects of each group interact with other group's Paro, they interacted with each group's Paro in different place in the facility. Moreover, we kept the existence of two kinds of Paro secret from subjects. Each groups interacted with each Paro about one hour at a time, four days a week for three weeks. We prepared a desk to set Paro in the center of people, and the subjects were arranged up as shown Fig.2. However, all the subjects couldn't interact with Paro at the same time. Therefore, we moved Paro among subjects in turn, and we made each subject's interaction time with Paro to be same.

3.2. Ways of evaluation

In order to investigate elderly people's moods before and after introduction Paro to the institution, the following two kinds of data and extra information were collected.
(1) Face scale [6] (Fig.3)
(2) Comments of nursing staffs
The Face Scale contains 20 drawings of a single face, arranged in serial order by rows, with each face depicting a slightly different mood state. A graphic artist was consulted so that the faces would be portrayed as genderless and multiethnic. Subtle changes in the eyes, eyebrows, and mouth were used to represent slightly different levels of mood. They are arranged in decreasing order of mood and numbered from 1 to 20, with 1 representing the most positive mood and 20 representing the most negative mood. As the examiner pointed at the faces, the following instructions were given to each patient: "The faces below go from very happy at the top

Table 1 Basic Attribute of 23 Subjects

	A	B
Total number of people	12	11
Male	4	2
Female	8	9
Age(AV±SD)	84.6±7.0	85.5±5.4

Fig.2 Interaction between Elderly People and Paro

to very sad at the bottom. Check the face which best shows the way you have felt inside now."

3.3. Results of evaluation

The face scale was applied to subjects, a week before introduction of Paro, 2nd and 3rd week after introduction.

As for face scale, we obtained data from 11 people of placebo Paro group, and from 7 people of regular Paro group. Fig.4 shows average face value. Average scores of regular Paro group decreased from about 9.0 (before introduction) to 7.0 (3rd week). Moreover, placebo Paro group's average scores also decreased from about 7.0 (before introduction) to 6.3 (3rd week). Therefore, interaction with regular and placebo Paro improved mood of subjects.

As for comments and observations of nursing staffs, both groups of subjects were waiting for Paro and participated interaction with Paro willingly. Paro increased their laughing, and encouraged subjects to communicate with each other and nursing staffs. In an interesting instance, an elderly man who was fastidious and difficult to communicate with other people, sang songs to Paro with big voice many times, and he made other people laughing. Another elderly made a song of Paro and sang it to Paro.

4. Discussions

We investigated the effects of Paro on elderly people who were staying in health service facility for the aged. Moreover, we compared the effects by the regular Paro with those by a placebo Paro. Against our expectation, face scale scores of regular and placebo Paro groups improved. From these results, regular and placebo Paro improved elderly people's moods.

Before experiment, we expected that people would lose interest in placebo Paro, because its reaction was very simple. However, subjects of placebo Paro group didn't lose interest in the placebo. They kept interaction with placebo Paro, and they didn't notice that placebo Paro's reaction was simple.

As for this fact, we consider following reasons:

(1) It was difficult for subjects to notice that placebo Paro's reaction was one pattern.

Subjects interacted with Paro in two or more people at the same time. Therefore, each subject's interaction time with Paro was not as long as they could notice that its reaction was one pattern.

(2) Reaction that cry and raise its head had special meanings.

Some subjects said "good boy" when Paro raised its head. They felt that Paro answered their calling.

In order to clarify these points, we will carry out experiments that use more number of Paro, and compare the effects of Paro, those of the placebo Paro and those of another placebo Paro that swing its head to right and left against stimuli.

5. Conclusions

We applied seal type mental commit robots, Paro to robot-assisted activity for elderly people at a health service facility for the aged. The experiment was carried out for 4 weeks in total. Then, we compared the effects by the regular Paro with those by a placebo Paro. The results show that interaction with regular and placebo Paro has psychological effect and social effect to elderly people.

We will have further experiments and research in different conditions and situations. Moreover, we will investigate relationship between functions of a mental commit robot and its effects to elderly people in robot-assisted activity.

References

[1] M. M. Baum, N. Bergstrom, N. F. Langston, L. Thoma, Physiological Effects of Human/Companion Animal Bonding, Nursing Research, Vol. 33. No. 3, pp. 126-129 (1984)

[2] 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)
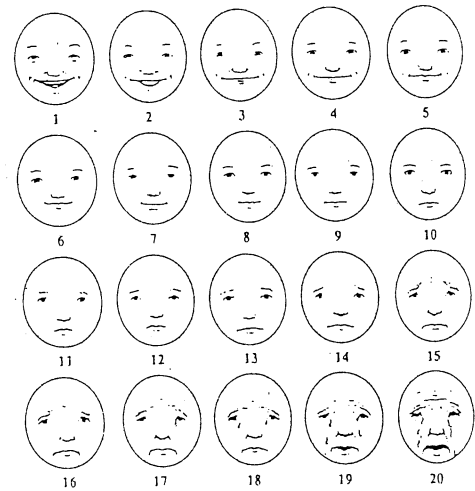
INSTRUCTIONS: The faces above go from very happy at the top to very sad at the bottom. Check the face which best shows the way you have felt inside now

Fig.3 Face Scale
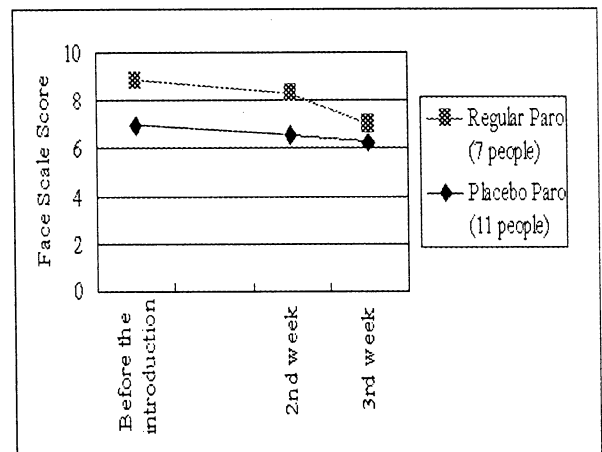
Fig.4 Average Face Scale Scores of Elderly People for 4 weeks

[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 (July. 2001) paper number 182 and 6 pages in CD-ROM Proc.

[4] K. Wada, T. Shibata, T. Saito, K. Tanie, Robot Assisted Activity for Elderly People and Nurses at a Day Service Center, Proc. of the IEEE Int'l Conf. on Robotics and Automation pp.1416-1421, 2002.

[5] T. Saito, T. Shibata, K. Wada, K. Tanie, 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 the 7th Int. Symp. on Artificial Life and Robotics Vol.1 pp.316-319, 2002.

[6] C. D. Lorish, R. Maisiak, The Face Scale: A Brief, Nonverbal Method for Assessing Patient Mood, Arthritis and Rheumatism, Vol. 29, No. 7, pp. 906-909 (1986)

An Intelligent Iterative Learning Controller Emulating Human Intelligence for Robotic Systems

M. Arif, T. Ishihara and H. Inooka

Graduate School of Information Sciences, Tohoku University, Sendai, JAPAN

Email: {arif,ishihara,inooka}@control.is.tohoku.ac.jp

Abstract

In many daily life activities, human acts as a control element very effectively under diverse environmental conditions and nonlinearities. In this paper, we have highlighted the human learning in repetitive tasks and tried to incorporate this aspect in iterative learning control. By introducing the element of intelligence in iterative learning control, it is possible to improve the performance of the controller without modifying its simple structure.

Keywords: Iterative Learning, Manual Control, Robotics, Trajectory Tracking

1   Introduction

A human performs many difficult tasks successfully and safely in his daily life activities. Driving an automobile, riding a bicycle, operating some machines in the industry are some of the examples. Each of these tasks involves a human being acting as a feedback element in control systems. Many researchers from past fifty years or more have modeled the human control motions. A good survey of these techniques can be found in [1]. Although it is very difficult to model the human decision-making, cognitive activities or human reasoning, the human performance in control of a complex non-linear system for some designated task can be modeled efficiently. Iterative learning is a common practice by a human being while performing routine tasks. An iterative manual control model of a human operator was proposed in [2] which describes human learning in the tasks of repetitive nature. The main features of this model are human experience/memory, information perception and extraction of useful information from the information available, decision-making and control. A human operator performs simple rule based control and modifies the control input in each repetition/iteration giving different weights to the information that he not only perceives directly but also processes the directly available information to get some more useful information. Based on this iterative manual control model of human operator, we have proposed an intelligent iterative learning controller for repetitive trajectory tracking tasks done by robotic manipulators in the robotic industry. We have incorporated the main features of human control to get better performance in the trajectory tracking tasks. We have highlighted the importance of the selection of the initial control input in a new trajectory tracking task based on the previous experience of the controller. Proper selection of the initial control input can effectively reduce the number of iterations required to reduce the tracking error below certain acceptable bound. The idea of generating initial control input based on the previous experience of controller is borrowed from the iterative manual control model of the human operator. Previous experiences of similar tasks done by human operator give him better guess of initial control input for a new task and he learns at faster pace. While human operator remembers his experiences in his brain, the experience of iterative learning controller can be
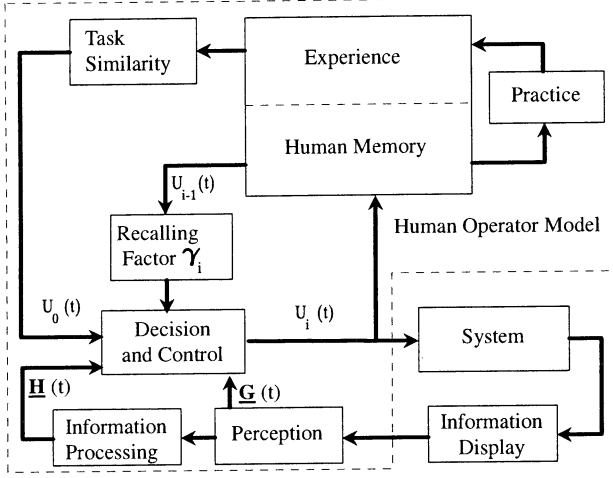
Figure 1: Iterative Manual Control Model

stored in the neural network trained on the data collected from the previous trajectory tracking tasks [3]. In another technique, the data of previous experience can also be stored in a database and data interpolation using local modelling technique can be used to predict the initial control input for a new desired trajectory [4]. Selection of the initial control input is independent of the structure of the controller; therefore this method is very general for all class of the iterative learning control schemes.

2 Iterative Manual Control Model

A human operator performs various control tasks in his daily life repeatedly and improves his performance based on the learning. This aspect of human learning and control was described by Arif et al. in [2]. Human learning and control is modeled as an iterative manual control model of human. This model is shown in the Figure 1. Given a certain tracking task, a human operator starts with some initial guess, if he did not encountered with this particular system and trajectory previously. If the human operator has an experience of controlling some similar task in the past,

he uses his past experience to generate a better initial guess. The human operator applies the initial control input to the system and gets the necessary information about the system by visual or other feedbacks. These informations can be error data or other mode of informations. He extracts some more information, that he feels necessary for the task, by processing the directly available information. Based on these information, he modifies his previous control action stored in his memory accordingly. This process repeats until the human operator reaches to a satisfactory control pattern in the sense of some norm. Once the human operator succeeds in getting the satisfactory control pattern, he improves his recalling factor and tries to master the task by practice: trying to regenerate the exact control pattern each time that he has found. After enough practice, his experience is modified accordingly. Major blocks of iterative manual control model are shown in figure 1 and explained in detail in [2].

In this model, it is assumed that the human operator gets the experience by practice or recall of his control actions and tries to make certain rules to control the system. The "experience/ human memory" and "decision and control" blocks of the model can be coupled to represents the decision making of human operator. The human memory can be classified into two broad categories, namely, short term or primary memory and long term or secondary memory. The control patterns or items stored in short term or primary memory are rapidly lost, if we do not perform any effort to maintain them, i.e., shifting them to the secondary memory by practice. The practice increases both speed and accuracy.

It is believed that the human operator processes the directly available information and extracts some useful information indirectly. He then uses both the informations that he perceived directly and indirectly in generating his control actions. The processing functions may be of taking derivative, taking integral or prediction the information in the future time. Human learning can be analogous to the iterative or repetitive control mentioned in [5].
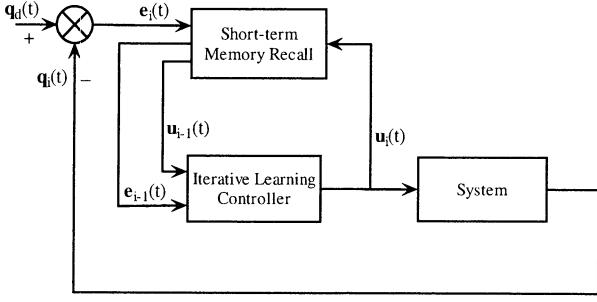
Figure 2: Basic configration of iterative learning control.

3 Iterative Learning Control

Iterative learning control algorithm was first proposed by Arimoto et al. [5] and can be described simply by Figure 2. In this algorithm the data or previous iteration only is stored in the short term memory. An iterative learning controller can be represented as,

$$\mathbf{u}_{i+1}(t) = \mathbf{u}_i(t) + f(\mathbf{e}_i(\mathbf{t}), \dot{\mathbf{e}}_i(t)) \qquad (1)$$

where $\mathbf{u}$ is the control input, $\mathbf{q}_d$ is the desired trajectory and $\mathbf{e}_i(t)$ is the tracking error in i^{th} iteration. The modification of control in iterative learning control is very much similar to the human learning in repetitive tasks [2]. Short term memory is used to recall the error and control information of the past iteration.

4 Emulating Human Intelligence

In the iterative learning control algorithm, the data of the last iteration or in some cases data of more than one past iterations is used in the construction of the control input for the current iteration. Data about the past iterations is stored in the computer memory. But as the iterations proceed, only the data of one or two past iterations, which are required in constructing the control input, is remembered. The past data
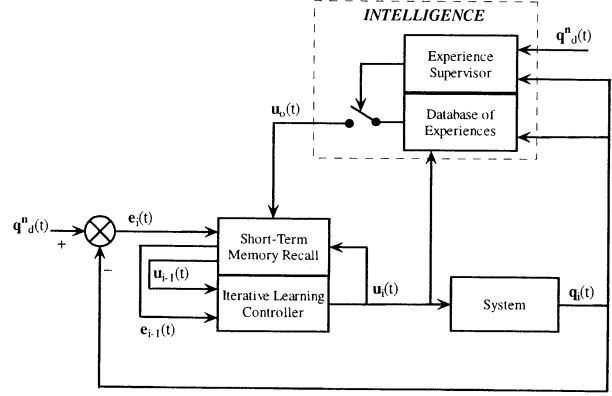
Figure 3: Intelligent iterative learning control

of the old iterations, which will not be used in the future iterations, is discarded. This mechanism works well if the task is bounded to a single desired trajectory tracking task. But if the task is to track multiple desired trajectories or the desired trajectory may change over the time, the learning that was acquired in the tracking task of the previous desired trajectory will be lost and for a new desired trajectory, iterative learning control algorithm will start from zero learning.

To overcome this problem, we have introduced a long term memory which is used to store the experiences of the iterative learning controller gained during various trajectory tracking tasks. This aspect is borrowed from iterative manul control model of human operator. The long term memory can be an experience database [4] or a neural network [3]. The data of previous experiences determines the region of the approximation $S \subset R^n$. For a new desired trajectory, let the set of desired states lies in the state space region S_d. The desired states region S_d should be a subset of S to ensure a good approximation. Experience supervisor keep the record of the state region S of the experience database. Whenever a new desired trajectory is presented for the tracking, the experience supervisor checks whether the state region S_d, in which the new desired trajectory lies, is a subset of the region S. If the new desired trajectory lies within the region S of the training pattern, then the

experience supervisor let the neural network or local modelling technique to approximate the initial control input $\mathbf{u}_0(t)$ for the new desired trajectory and provide this initial control input $\mathbf{u}_0(t)$ to the iterative learning controller. If the state region S_d is not the subset of S, the experience supervisor forces the iterative learning controller to start from the zero knowledge. Experience supervisor makes sure that the new trajectory lies within the subset of the experiences learned to avoid extrapolation which may cause large errors in the initial control input.

5 Application to Robotic Manipulators

The dynamic equation of a robotic manipulator having n degree of freedom can be described as follows,

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + G(\mathbf{q}) = \mathbf{u} \qquad (2)$$

where $\mathbf{q}(t) \in R^n$ is the angular joint positions, $M(\mathbf{q})$ is the inertia matrix, $C(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ is the Coriolis and centripetal forces, $G(\mathbf{q})$ are the gravitational torques and $\mathbf{u} \in R^n$ are the input torques applied to the joints of manipulators. Let the equation (2) can be rewritten as,

$$\mathbf{u}(t) = \mu(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \qquad (3)$$

The function $\mu(.)$ given in the above equation can be learnt by the neural network [3] or reconstructed through local modelling technique from the experience database [4]. This function can be very complicated and we do not expect that we can approximate it correctly for all input space. But from some input-output data we assume that we can approximate the function in a compact subset of input space. Each time when a new desired trajectory has to be tracked, an approximate desired control input $\widetilde{\mathbf{u}}_d(t)$ corresponding to desired states $(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)$ can be generated from the experience database which will then be used as an initial control input $\mathbf{u}_0(t)$ for iterative learning control mentioned in equation (1) in

a way that the control input $\mathbf{u}_i(t)$ converges to the desired control input $\mathbf{u}_d(t)$ as number of iteration increases. Some simulations of single link and two link robot manipulators were done in [3] and [4] to confirm the improvement in the convergence of tracking error when we use intelligent iterative learning controllers.

6 Conclusions

In this paper, an aspect of human intelligence is included in iterative learning control algorithms to improve their performance in trajectory tracking tasks. Previous experience of the iterative learning controller can be used to predict the initial control input for new tracking tasks. This will help the tracking error to converge faster to an acceptable error bounds.

References

[1] Hess R. (1997), Feedback control models - Manual control and tracking. In Handbook of human factors and ergonomics, 2nd edition, Ed. by Salvendy G., John Wiley and Sons Inc., pp. 1249-1294.

[2] Arif M. and Inooka H. (1999), Iterative manual control model of human operator. Biological Cybernetics, Vol. 81, pp. 445-455.

[3] Arif M., Ishihara T. and Inooka H. (2002), Generalization of Iterative learning control for Multiple Desired Trajectories in Robotic Systems, PRICAI 2002: Trends in Artificial Intelligence, Lecture Notes in Artificial Intelligence, Edited by M. Ishizuka and A. Sattar, Springer-Verlag, pp. 295-304.

[4] Arif M., Ishihara T. and Inooka H. (2001), Incorporation of experience in iterative learning controllers using locally weighted learning", Automatica, Vol 37(6), pp. 881-888.

[5] Arimoto S., Kawamura S. and Miyazaki F. (1984), Bettering operation of robots by learning, J. Robotic Systems, Vol. 1(2), pp.123-140.

Analysis of Human Walking Gait of Young and Elderly Subjects using Detrended Fluctuation Analysis Technique

Muhammad Arif*, Yasuaki Ohtaki**, Tadashi Ishihara* and Hikaru Inooka*

*Graduate School of Information Sciences, Tohoku University, Aramaki aza 01, Aoba-ku, Sendai

**Graduate School of Engineering, Tohoku University, Aramaki aza 01, Aoba-ku, Sendai

Abstract

In this paper, we have analyzed the acceleration of center of gravity data of young and elderly subjects during normal walking using detrended fluctuation analysis technique. It was observed that the scaling exponents of elderly subjects in short term region are less than young subjects. It shows that walking of eldelry subjects is less correlated in the short term region puting them on more risk of falls.

Keywords: Walking gait, Center of gravity, detrended fluctuation analysis

1 INTRODUCTION

Detrended fluctuation analysis (DFA) was proposed by Peng et al.[1] in 1994 to detect the long range correlations in the time series having non-stationarities. It estimates long range power law correlation exponents in the noisy signals of various physical and biological systems. DFA provides a simple parameter called scaling exponent α to characterize the complex behavior of the system in terms of long range correlations. It was used as a method of correlation analysis in the diverse research fields from biological and physiological systems to financial time series[2,3,4,5]. DFA is based on the idea of random walk theory [6]. Detrending of the time series is used to remove the non-stationary components. Time series is mapped onto a self similar series through simple integration. This

technique avoids wrong detection of correlations due to the presence of non-stationarity. This method can also be used to differentiate between various states of behavior of a particular system at different times. We have used DFA in the same context to differentiate between the walking gait data of young and elderly people.

Many researchers have studied human walking gait and several studies have been reported about the change in the kinematics parameters with age [7,8]. Movement of center of gravity (COG) of a person during walking can be an important index of walking stability of young and elderly subjects because during walking we tries to control our COG to prevent fall. Acceleration of COG data is recorded using a light weight portable measurement system and is analyzed using DFA method to observe the effect of age on the correlations present in the data.

2 Detrended Fluctuation Analysis (DFA)

Let a time series $x(t)$ is represented as x_i, $i = 1, 2, \ldots, n$ at discrete time steps. This time series can be mapped to a self similar process by simple integration,

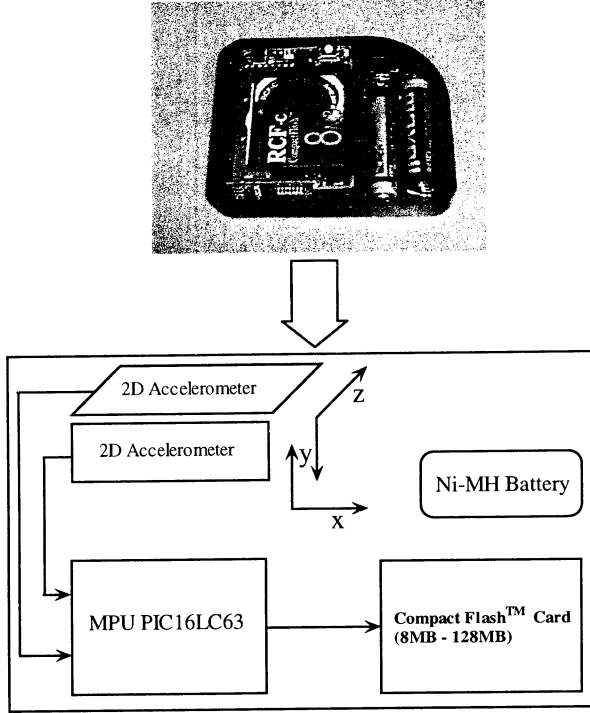$$y(k) = \sum_{j=1}^{k} (x_j - \bar{x}) \qquad (1)$$

Figure 1: Portable 3D acceleration measurement system

Figure 2: Position of Sensor Unit

where $\bar{x}$ is the mean of the time series x. The time series $y(k)$ is then divided into N small non-overlapping boxes of equal length s. In each box, an straight line is fitted in least square sense that shows the local trend of the time series $y(k)$ in small segment of length s. Let this local trend can be represented as y_ν in ν^{th} segment. The variance function $F_s(\nu)$ can be calculated for ν^{th} segment as,

$$F_s^2(\nu) = \frac{1}{s} \sum_{l=1}^{s} [y((\nu-1)s+l) - y_\nu(l)]^2 \quad (2)$$

Finally detrended fluctuation function $F(s)$ associated with the box size s can be calculated as,

$$F(s) = \sqrt{\frac{1}{N} \sum_{\nu=1}^{N} F_s^2(\nu)} \quad (3)$$

$F(s)$ is measured for s ranging from 4 to $n/4$. If

the time series $x(j)$ has long range power correlations then fluctuation function will follow the following law,

$$F(s) \propto s^\alpha \quad (4)$$

where the scaling exponent α can be calculated by plotting $F(s)$ vs s on log-log scale and calculating the slope of the least square line fitted on the curve. Value of scaling exponent determines the type of correlations. Scaling exponent of un-correlated white noise will be 0.5 whereas scaling exponent more than 0.5 shows positive correlations. Scaling exponent less than 0.5 will show anti-correlated behavior.

3 EXPERIMENTAL SETUP

Twelve Subjects have participated in the experimental study and are divided into two groups. Group YNG consists of six young healthy subjects having ages between 21 to 30 years (mean 25 ± 4 years), body weight distribution was 67 ± 6 kgs and height distribution was 174 ± 4 cms. Group OLD consists of six elderly healthy subjects with no major disease having ages between 60 to 80 years (mean 72 ± 4 years), body weight distribution is 61 ± 9 kgs and height distribution is 157 ± 12 cms. All subjects were asked to walk normally on $20m$ straight walking track. Ac-

Figure 3: Fluctuation function $F(s)$ for young subjects in all three directions

	Lateral	Vertical	Ant/post
Young	1.53 ± 0.06	1.42 ± 0.09	1.41 ± 0.09
Elderly	1.37 ± 0.16	1.26 ± 0.14	1.12 ± 0.13

Table 1: Scaling exponent in Region I for Young and Elderly subjects

	Lateral	Vertical	Ant/post
Young	0.43 ± 0.14	0.06 ± 0.01	0.08 ± 0.02
Elderly	0.44 ± 0.15	0.04 ± 0.009	0.06 ± 0.02

Table 2: Scaling exponent in Region II for Young and Elderly subjects

4   Experimental Results

Accelerations of COG in all three directions, i.e. lateral, vertical and anterior/posterior, are analyzed using detrended fluctuation analysis technique. Fluctuation function $F(s)$ is plotted versus s for lateral, vertical and anterior/posterior directions in Figure 3. Points in grey sho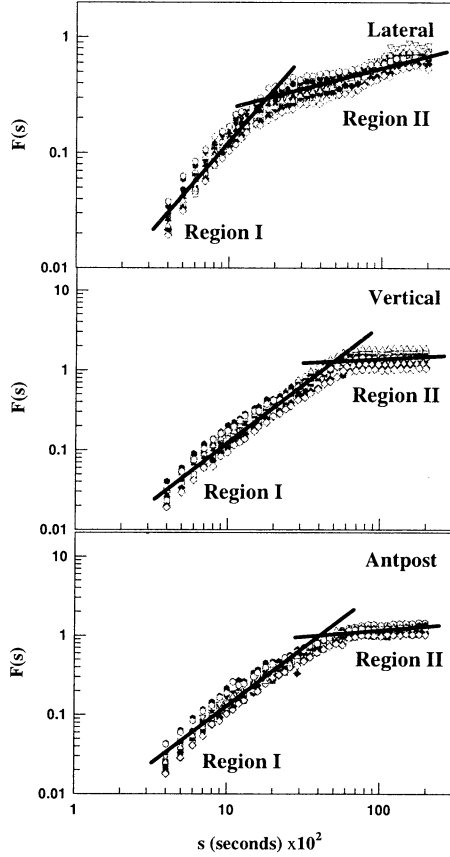ws the values of fluctuation function $F(s)$ for six young subjects while straight lines show the trend of $F(s)$. It is obvious from the figure that the fluctuation function divides into two regions showing different trends in these regions.

Crossover point of the two regions is the point where two trends cross each other. Shape of the fluctuation function $F(s)$ in vertical and anterior posterior directions suggest the presence of periodic behavior [9]. This observation is very obvious as human walking is a periodic activity. Scaling exponent α in region I (short range) for young and elderly subjects in all three directions are given in Table 1. It can be observed that the scaling exponents are reduced ($p < 0.01$) for elderly subjects in all three directions as compared to young subjects. Hence the acceleration data in elderly subjects are less correlated as compared to young subjects. Crossover points between region I and II for young subjects are 0.16 ± 0.02, 0.53 ± 0.03 and 0.47 ± 0.02 seconds in lateral, vertical and anterior/posterior directions respectively. Whereas crossover points between region I and II for elderly subjects are 0.17 ± 0.03, 0.48 ± 0.04

celeration of the COG during walking was measured by a portable 3D acceleration measurement system made by ITR Co. Ltd.(Japan), shown in Figure 1. 3D acceleration is measured by two 2D acceleration sensors that can sense accelerations up to $\pm 2g$. The 3D acceleration measurement system was placed on the trunk at about 55% of the subject's height using a waist belt as shown in the Figure 2. Acceleration of the COG was recorded in lateral, vertical and anterior/ posterior directions in Flashcard memory and later transferred to computer for analysis. In Figure 2, x direction refers to lateral direction, y direction refers to vertical direction and z direction refers to anterior/ posterior direction. The data from the accelerometer was sampled at the frequency of 100Hz.

and 0.42 ± 0.04 seconds in lateral, vertical and anterior/posterior directions respectively. Please note that the average walking step time of young subjects is 0.55 ± 0.02 seconds and average walking step time of elderly subjects is 0.46 ± 0.03 seconds. So crossover points in vertical and anterior/posterior directions represent the approximate time period of the periodic walking gait signal. But human does not keep the lateral postural sway periodic during walking. In region II of lateral acceleration, both young and elderly subjects have shown the scaling exponent α near 0.44 which is slightly less than 0.5. This suggests that acceleration of COG in lateral direction is slightly anti-correlated. Vertical and anterior/posterior scaling exponents show point-wise correlations like random walk or Brownian noise ($\alpha = 1.5$) in young subjects. Reduction of scaling exponent in short term region in elderly subjects shows loss of correlations in the acceleration data. In region II, both vertical and anterior/posterior acceleration have shown very small scaling exponent (anti-correlations) typical to periodic signals or sinusoidal trends [9].

5   Conclusions

We have studied the behaviors of the acceleration of center of gravity of human body during normal walking in young and elderly subjects. Elderly subjects have shown significant reduction ($p < 0.01$) in the scaling exponents of lateral, vertical and anterior/posterior directions in short term region (within one walking step time) as compared to young subjects. Whereas no significant change was observed in the long term region (time greater than one walking step) in young and elderly subjects. This shows that the walking of elderly subjects are less correlated in the short term region.

References

[1] Peng C.K., Buldyrev S.V., Havlin S., Simons M., Stanley H.E., and Goldberger A.L. (1994), Mosaic organization of DNA nucleotides, Physical Review E, Vol. 49(2), pp. 1185-1689.

[2] Absil P.A. ,Sepulchre R. ,Bilge A. and Gerard P. (1999), Nonlinear analysis of cardiac rhythm fluctuations using DFA method, Vol. 272(2), pp. 235-244.

[3] Peng C.K., Hausdorff J.M., Havlin S., Mietus J.E., Stanley H.E. and Goldberger A.L. (1998), Multiple-time scales analysis of physiological time series under neural control, Physica A: Statistical and Theoretical Physics, Vol. (249), pp. 491-500.

[4] Liu Y., Cizeau P., Meyer M., Peng C.K. and Stanley H.E. (1997), Correlations in economic time series, Physica A: Statistical and Theoretical Physics, Vol. 245, pp. 437-440.

[5] Yoshinaga H., Miyazima S. and Mitake S. (2000), Fluctuation of biological rhythm in finger tapping, Physica A: Statistical and Theoretical Physics, Vol. 280, pp. 582-586.

[6] Shlesinger M.F., West B.J., and Klafter J. (1987), Levy dynamics of enhanced diffusion: Application to turbulence, Physical Review Letters, Vol. 58, pp. 1100-1103.

[7] Winter D.A., Patla A.E., Frank J.S. and Walt S.E. (1990), Biomechanical walking pattern changes in the fit and healthy elderly persons, Phys. Ther., Vol. 70, pp. 340-347.

[8] Ostrosky K.M., Vanswearingen J.M., Burdett R.G. and Gee Z. (1994), Comparison of gait characteristics in young and old subjects, Phys. Ther., Vol. 74, pp. 634-646.

[9] Hu K., Ivanov P.C., Chen Z., Carpena P. and Stanley H.E. (2001), Effect of trends on detrended fluctuation analysis, Physical Review E, Vol. 64, 011114 (19 pages).

Solving Multi-time Period Production/Distribution Problem by Using Spanning Tree-based Genetic Algorithm

Mitsuo Gen
Department of Industrial &
Information Systems Engineering,
Ashikaga Institute of Technology,
Ashikaga 326-8558, Japan
E-mail: gen@ashitech.ac.jp

Hiroyuki Nozawa
Department of Industrial &
Information Systems Engineering,
Ashikaga Institute of Technology,
Ashikaga 326-8558, Japan

Admi Syarif
Department of Industrial &
Information Systems Engineering,
Ashikaga Institute of Technology,
Ashikaga 326-8558, Japan
Department of Mathematics,
Lampung University,
Bandar Lampung, Indonesia, 35148

Abstract: The production, distribution and inventory problem with multi-time periods has recently been receiving the attention and the interest of the researchers because of its great use in public distribution or logistics system. As the design systems of facilities have grown even more complex, most of production/distribution planning problems become more computationally difficult. This paper deals with a multi-time period production/distribution problems. By assuming that the plant capacities and the customer demand are known in advance, we examine how the distribution centers should be designed and how the product should be produced and transported. We propose a new technique called hydridized spanning tree-based genetic algorithm (hst-GA) to solve the problem. We utilize Prufer number to represent the candidate solution to the problem. In order to improve the performance of the algorithm, a new local search technique called displacing Prufer number is included into the proposed method. Local search here is designed to help move out of a local optimum and carefully search the near optimal region. We compare the numerical experimental results of the proposed method with those of traditional method to demonstrate the effectiveness of the proposed method.

Keywords: production/distribution problem, logistics, Genetic Algorithm, Prufer number, local Search

1. Introduction

To be competitive in the global manufacturing environment, the strategic production/distribution planning from time to time becomes one of the most important factors. This planning problem exists because there are limited production resources that cannot be stored from period to period. There are large numbers of literature dealing with various variation production/distribution problems. However, in most case, they consider an overall production strategy, inventory strategy and flow of products through a facility over in a single period of time in order to minimize cost or maximize profit. The first work of Geoffrion and Graves solved a multi-commodity location problem that optimized flow of product over the plant-to-customer path via a warehouse [1].

Since the invention of the GA by John Holland in 1975 [2], it has been demonstrated to be a versatile and effective approach for solving many diverse and difficult problems. The GA is one of a family of heuristic optimization techniques, which include simulated annealing, Tabu search, and evolutionary strategies. For the specific engineering problems, however, GA cannot be guaranteed the optimality and sometimes can suffer from the premature convergence situation of its solution because of their fundamental requirements that are not using a priori knowledge and not exploiting local search

information [3]. In order to overcome this problem, several hybridized technique were introduced. The most popular technique is to combine the local search technique into GA process. Here, genetic search is used to perform global exploration among the population and local search is used to perform local exploitation around chromosomes.

Another important issue in GA is how to determine the best GA parameters that can bring us to the optimal solution. To this problem, fuzzy logic controller (FLC) is acceptable because only incomplete knowledge and imprecise information are available for identification of the relation between the GA parameters and the behavior of GA. FLC provides an algorithm that can convert linguistic control strategy based on expert knowledge into an automatic control strategy [4,5]. Adapting such GA parameters automatically can not only improve the quality of solution but also reduce the computational time.

In this paper, we concern with one of production/distribution planning problem that has multi-time period. Our purpose here is to introduce a new technique for designing an efficient production/distribution and inventory planning so that products are produced and distributed at the right quantities, to the right customers, and at the right time, in order to minimize system wide costs while satisfying all demand required. This problem can be viewed as an optimization model that integrates facility location decisions, transportation/distribution costs, and inventory management for multi-products and multi time periods. In this problem, we consider multiple products with independent demand, multiple shared resources and multi-time periods. This problem is known as one class of network model. In the following figure 1, we show an illustration of the problem discussed in this paper. The subscript t here indicates the time period.
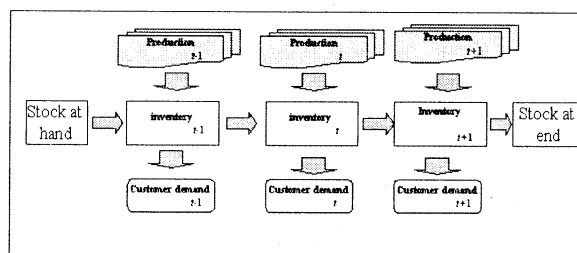
Figure 1. The illustration of the problem

To solve this problem, we propose a novel technique called hst-GA. We focus on this special graph and utilize the Prufer number [6] encoding based on a spanning tree which is adopted as it is capable of equally and uniquely representing all possible trees. We have shown in our previous research that this Prufer number representation, for some network problems

including transportation problem, gives better performance that traditional matrix-based representation. Further, to improve the performance of the algorithm, we adopt the automatic fine-tuning for the crossover ratio and mutation ratio using FLC [5]. In order to carefully search in the area of the optimal solution, in our proposed method, we also developed a new local search technique called displacing Prufer number and combined it into GA process. The effectiveness and efficiency of the proposed method are evaluated by comparing the numerical experiment results of the proposed method with those of traditional methods.

The rest of this paper is organized as follows: The mathematical formulation of this problem is given in Section 2. In Section 3, we describe the design of our algorithm including the chromosome representation, the GA process and FLC concept for auto-tuning the GA parameters. In Section 4, Numerical experiments and comparison with the results of traditional algorithm are presented to demonstrate the efficiency of the proposed method. Finally, some concluding remarks are given in Section 5.

2. Mathematical Model
In this section, we present a comprehensive mathematical model that considers real-world factors and constraints of the problem. Firstly, the notations used in the model are defined. The mathematical model with its objective function and constraints will be expressed later.

Indexes
t index of time period
i index of product
j index of plant
k index of resource
m index of customer

Parameters
a_{ijk} amount of resource k required to produce one unit of product i at plant j.
$b_{jk}(t)$ amount of resource k available at plant j in period t.
$d_{im}(t)$ demand for product i by customer m in period t.
$p_{ij}(t)$ unit cost of production for product i at plant j in period t.
$q_{ij}(t)$ unit inventory holding cost for product i at plant j in period t.
$c_{ijm}(t)$ shipping cost of product i from plant j to customer m in period t.

Variables
$x_{ij}(t)$ the production of product i at plant j during the time period t.
$y_{ij}(t)$ the inventory of product i at plant j at the end of time period t.
$z_{ijm}(t)$ the number of product i to be shipped from plant j to customer m in the time period t.

In this problem, we want to determine the production number for each product in each plant, inventory strategy and distribution network design to satisfy the resource capacity and customer demand with minimum cost. It can then be formulated as follows:

$$\min \sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J} p_{ij}(t)x_{ij}(t) + \sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J} q_{ij}(t)y_{ij}(t) + \sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{m=1}^{M} c_{ijm}(t)z_{ijm}(t) \quad (1)$$

subject to:

$$y_{ij}(t-1)+x_{ij}(t)-\sum_{m=1}^{M}z_{ijm}(t)=0, \quad \forall i, j, t \quad (2)$$

$$\sum_{j=1}^{J}z_{ijm}(t)=d_{im}(t), \quad \forall i, m, t \quad (3)$$

$$\sum_{i=1}^{I}a_{ijk}x_{ij}(t)\leq b_{jk}(t), \quad \forall j, k, t \quad (4)$$

$$x_{ij}(t),\; y_{ij}(t),\; z_{ijm}(t)\geq 0, \quad \text{for all } i, j, m, t \quad (5)$$

In the above model, the objective function captures production and inventory holding cost, which depend on the plant, plus transportation or distribution cost for shipment of product from plant to the customer. The constraint (2) is the inventory balance constraint that assures the supply of an item at each plant is either held in inventory or shipped to a customer to meet demand. The constraint (3) ensures that the shipments satisfy the demand of each customer for each period. The constraint (4) is a set resource constraint. Production in each period is limited by the availability of a set of shared resources. Typical resources are various types of labor, process and material handling equipment and transportation modes.

3. Design of the Algorithm
In this section, we shall propose our basic concepts and searching procedures for the proposed method (hst-GA). Firstly, we shall present the GA procedure. Secondly, the local search technique for finding a better solution with in GA loop is proposed. Finally, the FLC concept to make auto-tuning of the GA parameters is proposed.

3.1. Spanning Tree-based Genetic Algorithm
3.1.1 Representation
The genetic representation is a kind of data structure which represents the candidate solution of the problem in coding space. It should translate the under lying problem as accurately as possible into a number of code which can be manipulated by using the genetic operations. Usually different problems have different data structures or genetic representations. For this problem, solution including its transportation graph is represented as a spanning tree. Basically, there are three ways of encoding spanning tree: 1) edge encoding, 2) vertex encoding and 3) edge-and vertex encoding [7].

In 1889, Cayley [8] proved that the number of spanning tree in a complete graph of N nodes is equal N^{N-2}. Prufer [9] gave the simple proof of Calley's formula by establishing a one correspondence between the set of spanning and N-2 digit integer number with each integer between 1 and N inclusive [10]. The N-2 digit number later is known as Prufer number. This Prufer number encoding procedure is belong to the class of vertex encoding. It is shown by Gen that Prufer number encoding not only is capable of equally and uniquely representing all possible spanning tree, but also explicitly contains the information of node degree. Any node with degree d will appear exactly d-1 times in the encoding. This means that when a node appears d times in Prufer number, the node exactly have d+1 connections with other node. For a complete graph with p nodes, there are p^{p-2} distinct labeled trees. So we can use only permutation of p-2 digits in order to uniquely represent a tree with p nodes where each digit is an integer between 1 and p inclusive. When generating the Prufer number, however, it is also feasible for us to generate an infeasible Prufer number that cannot be adapted to generate the transportation/distribution graph. Gen [10] developed the feasibility criteria for Prufer number to be decoded into a spanning tree, however, we found that

their criteria is very complicated. In our previous work, we have designed a simple feasibility check criteria and repairing procedure for the Prufer number to be decoded into spanning tree in [11]. After checking its feasibility, from a Prufer number, a unique distribution graph is possible to be generated by using procedure given in [10, 12].

After decoding all Prufer numbers, we can determine the necessity of production for each product in each plant for each time period. In this sense, if there exists remaining resources in the plants, they can be used for manufacturing several numbers of the products and put them as the inventory.

3.1.2 Initialization
As the initial population, we generate *pop_size* chromosome. Each individual chromosome in the initial population is a candidate solution to the problem. We use the complete random method to generate the initial population.

3.1.3. Genetic Operators
Crossover: the one-cut-point crossover operation is used. For avoiding unnecessary decoding from which an infeasible chromosome (a Prufer number) may be generated after crossover operator, we add the criterion for feasibility of the chromosome.

Mutation: the inversion mutation is used. The inversion mutation is selects two positions within a chromosome at random and then inverts the sub-string between these two positions. This mutation operator always generates feasible chromosome if the parents are feasible, because the feasibility criteria is unchanged after these operations.

3.1.4 Evaluation and Selection
In the optimization problems, usually use of the objective function as the evaluation function to evaluate the fitness of chromosomes. For simplify, we taken the cost function of this problem as fitness function.
It is given as follows:

Procedure: evaluation

Step 1. decode all Prufer numbers into spanning trees;
Step 2. determine over all production cost, inventory cost and transportation cost;

In the selection procedure, we use the mixed strategy with $\mu+\lambda$-selection and roulette wheel selection to enforce the best chromosomes into the next generation. This strategy selects μ best chromosomes from μ parents and λ offspring. If there are no μ different chromosomes available, then the vacant pool of population is filled up with roulette wheel selection.

3.2. Local Search Technique
Though it has been shown that GA is an effective way to find the solution for many difficult optimization problems, when searching the space of possible solutions, we often reach a local optimum and find it difficult to make further progress. To help move out of a local optimum and carefully search the near optimal region, several researchers combine GA with local search technique.

Many researchers had reported that combining a GA and a local search technique into a hybrid approach often produces certain benefits [13]. This is because hybrid approach can combine the merits of a GA with those of a local search technique. That is, by applying GA, hybrid approach is less likely to be trapped in a local optimum

than a local search technique. In hybridizing GA using local search techniques, most common form is to incorporate a local search technique in GA loop [10, 14].

In this paper, we develop a simple new local search technique called displacing Prufer number. The displacing Prufer number procedure is as follows:

Procedure: local search

Step 1. select one digit in the Prufer number randomly;
Step 2. exchange it with the first Prufer number;

The Prufer number resulted by this operation is guaranteed to be feasible since the appearance number for each node remains the same.

3.3. Fuzzy Logic Controller
When using GA, one of the important factors is a balance between exploitation and exploration in the search space. To provide this balance, determination of design strategy for GA parameters such as population size, maximum generation, crossover probability and mutation probability is one critical issues. To solve this problem, several researchers have reported the use of FLC to automatically tune the GA parameters (See for example: 4, 5). The main idea of FLC is to dynamically change the GA parameters based on the information in the previous generations such as the average fitness of the population.

In our implementation of fuzzy logic controller, we make modification on Wang *et al.*'s concepts [5] to regulate automatically the GA parameters, crossover ratio p_C and mutation ratio p_M. The heuristic updating principles for the crossover and mutation ratio are to consider changes in the average fitness of the populations. As the inputs to the crossover fuzzy logic controller are changes in average fitness at consecutive two generations and the output is the change in crossover ratio $\Delta c(t)$. Based on a number of experiment and domain expert opinion, the input values are respectively normalized into the range [-4.0, 4.0] according to their corresponding maximum/minimum values.

For simplicity, we use a look-up table for fuzzy logic controller action as given in [5]. Using that look-up table, we found the value of scaling value $Z(i, j)$.
To get the output of this fuzzy logic scheme, we use the following formula:

$$\Delta c(t) = 0.02 \times Z(i, j) \qquad (6)$$

where $i, j \in \{-4, -3, -2, -1, 0, 1, 2, 3, 4\}$.

After getting the output of this fuzzy logic controller system, then the crossover ratio for the next generation is regulated as follows:

$$p_C \ (t+1) = p_C(t) + \Delta c(t) . \qquad (7)$$

We use the similar concept to regulate the mutation ratio. However the control action is done using the following formula:

$$\Delta m(t) = 0.002 \times Z(i, j) \qquad (8)$$

where $i, j \in \{-4, -3, -2, -1, 0, 1, 2, 3, 4\}$.

4. Numerical Experiments
The proposed hst-GA was implemented in visual C language and run on PC Pentium 700MHz. To confirm the effectiveness of the proposed method, we tested our proposed method using two different sizes of test problems as given in Table 1. The necessity of part for producing each product is given in the Table 2.

Table 1. Design of the test problems

Problem	I	J	K	M	T
1	3	4	2	7	2
2	3	6	2	21	2

Table 2. The necessity of parts for producing products

Part/Product	1	2	3
1	1.8	1.5	0
2	1.2	1.4	2.0

In this numerical experiment, we randomly generate the production, inventory and distribution cost for each product in the range [3]. The costs for the second time period are considered to be 1.2 times of the cost in the first time period. For each test problem, we had two numerical experiments by giving different population size and maximum generation. Each numerical experiment was run 10 times. We set the initial p_C and p_M as 0.4 and 0.2 respectively. We compared the numerical experiment results of the proposed method with those of traditional spanning tree-based genetic algorithm. Table 3 summarizes the results of the numerical experiments. To show the effectiveness of the proposed method, in the following figure 2, we illustrate the convergence behavior of the proposed method in comparison with the traditional method.
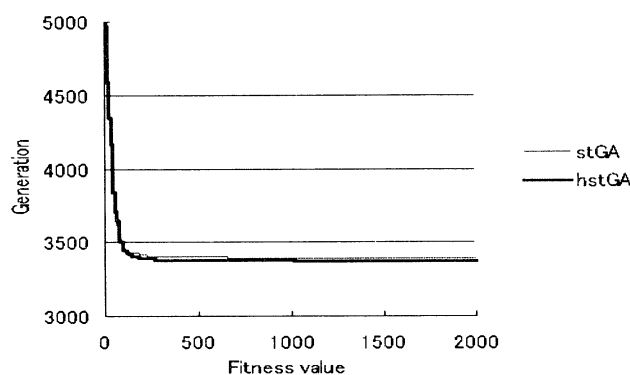
Figure 2. The fitness value in the generation

5. Conclusion

In this paper, we proposed a novel technique called hst-GA to solve the multi-time period production/distribution problem. We adopt the Prufer number as it equally represents a spanning tree. In order to improve the effectiveness of genetic algorithm, we develop a local search technique and included into GA loop. Moreover, FLC was also hybridized to the evolutionary process for making auto-tuning of the GA parameters. We carried out several numerical experiments by both hybridized spanning tree-based genetic algorithm and traditional spanning tree-based genetic algorithm to demonstrate the effectiveness and efficiency of the proposed approach. It has been shown that the proposed method gives better results.

References

[1] Graves, S. C. (1999) Manufacturing Planning and Control, MIT working paper, http://www.mit.edu/sgraves

[2] Holland, J. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.

[3] Lee, C. Y., Yun, Y. S. and Gen, M. (2002) Reliability Optimization Design for Complex System by Hybrid GA with Fuzzy Logic Controller and Local Search, *IEICE Trans. Fundamental*, vol. E85-A, no.4, pp. 880-891, 2002.

[4] Lee, M. and H. Takagi, ``Dynamic control of genetic algorithm using fuzzy logic techniques," *Proc. of the 5th Inter. Conf. on Genetic Algorithm*, 1985.

[5] Wang, G. S. Wang, and Z. G. Hu, ``Speeding up the Search Process of Genetic algorithm by Fuzzy Logic," *Proc. the 5th European Congress on intelligent Techniques and Soft Computing*, pp. 665-671, 1997.

[6] Dossey, J., A. Otto, L.spence and C. Eynden, *Discrete Mathematics*, Harper Collins, 1993.

[7] Lo, C. C. and Chang. W. H. (2000), A multiobjective hybrid genetic algorithm for the capacitated multipoint network design problem, *IEEE Transaction on System, Man and Cybernetic*, Vol. 30, No. 3, pp.461-470.

[8] Cayley, A. (1889) A Theorem on Tree, *Quart. J. Math.*, Vol. 23, pp. 376-378.

[9] Prufer, H. (1918) Neuer bewis eines saizes uber permutationen, *Arch. Math. Phys,*

[10] Gen, M. and R. Cheng. (1997) *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, Vol, 27, pp. 742-744.

[11] Syarif, A. and Gen, M. (2002), Solving Exclusionary Side Constrained Transportation Problem by Using A Hybrid Spanning Tree-based Genetic Algorithm, *Journal of Intelligent Manufacturing*, to appear.

[12] Gen, M. and R. Cheng (2000) *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York.

[13] Mathias, K. E. L. D. Whitley, C. Stork, and T. Kusuma, ``Staged hybrid genetic search for seismic data imaging," *Proc. of the Evolutionary Computation*, pp. 356-361, 1994

[14] Michalewicz, Z., *Genetic Algorithms + Data structure = Evolution Programs*, Springer-Verlag, 1992; Second, extended ed., 1994; Third, revised and extended ed., 1996

Table 3. Computational results

	pop_size	max_gen	st-GA			hst-GA		
			best	Average	ACT	best	Average	ACT
1	50	2000	3371.0	3394.2	43.7	3371.0	3375.2	52.4
	75	3000	3371.0	3394.2	63.2	3371.0	3375.2	78.2
2	75	3000	16732.0	16823.7	94.8	16687.0	16783.5	107.2
	100	5000	16687.0	16736.3	164.3	16687.0	16714.2	175.3

ACT : Average computational Time (in second)

Network-based Hybrid Genetic Algorithm
to the Scheduling in FMS Environments

KwanWoo Kim* Mitsuo Gen**

* Department of Intelligent Systems
Tokyo Metropolitan Institute of Technology,
Hino-city, Tokyo, 190-0065, Japan
e-mail: kwkim@genlab.ashitech.ac.jp

LinLin**Genji Yamazaki*

**Department of Industrial & Information Systems
Engineering, Ashikaga Institute of Technology,
Ashikaga 326-8558, Japan
e-mail: gen@ashitech.ac.jp

Abstract

Scheduling in flexible manufacturing systems (FMS) must take account the shorter lead-time, multiprocessing environment, the flexibility of alternative workstations with different processing times, and the dynamically changing states. The best scheduling approach described in this paper is to minimize makespan t_M, total flow time t_F, and total tardiness penalty p_T. But in the case of the manufacturing system problems, it is difficult for those traditional optimization techniques to cope with. This paper presents a new flow network-based hybrid genetic algorithm (hGA) approach for generating static schedules in a FMS environment. The proposed method combines with the neighborhood search technique in mutation operation to improve solution of the FMS problem. The numerical experiments show that the proposed flow network-based hGA is both effective and efficient to the FMS problems.

Keywords: Flexible Manufacturing Systems (FMS), Genetic Algorithms (GA), and Scheduling.

1. Introduction

Development of job processing schedules in a flexible manufacturing system (FMS) environment is a complex task. The FMS consists of a set of workstations capable of performing a number of different operations and interconnected by transportation jobs. All of the algorithms used to optimize this class of problems have an exponential time, that is, the computation time increases exponentially with problem size. There has been a surge of research interest in applying genetic algorithms (GAs) to the problem [4][6][8].

More recently, heuristic algorithms to find the optimal solution like simulated annealing (SA), GA and tabu search (TS) have emerged [2][8]. The SA algorithm generates a new sequence from an existing sequence by making history-independent, random changes to the existing sequence. The TS algorithm also operates on a single existing sequence, but attempts to learn from experience by maintaining a list of tabu 'moves. In contrast, the GA method operates on multiple sequences simultaneously. It seeks to maintain a population of several good sequences at all times, based on past experience [8]. The growing interest in GAs and their capabilities led to their consideration as possible means to solve scheduling problems in FMS. GA-based approaches can effectively combine the problem processing knowledge with rote-learned knowledge to generate good quality schedules [8]. However, a pure GA-based approach requires specially designed GA operators and is liable to generate illegal schedules.

Yang [6] presented GA-based dynamic programming approach for generating static schedules in a FMS environment. However, the major weakness of the models introduced so far lies in that he considered the alternative machines for each operation with a fixed sequence when constructing a schedule.

This paper presents a new flow network-based hybrid genetic algorithm (hGA) approach for generating static schedules in a FMS environment. The proposed method combines with the neighborhood search technique in mutation operation to improve solution of the FMS problem which minimizes makespan t_M, total flow time t_F, and total tardiness penalty p_T.

The rest of the paper is organized as follows. In Section 2, we describe a hypothetical instance of FMS problem and the mathematical models of this problem are given in Section 3. In Section 4, we describe the feature of our method including the chromosome representation, the hybrid genetic algorithm process. In Section 5, numerical experiments are presented to demonstrate the efficiency of the proposed method. Finally, some concluding remarks are given in Section 6.

2. Flexible manufacturing systems (FMS) problem

Flexible Manufacturing System (FMS) is an enhancement of the cellular manufacturing paradigm, and represents the state-of-the art in the design of manufacturing systems. Typical flexible manufacturing system is composed of multiple workstations (or machine centers), a material handling system, and a loading-unloading station [1].

Figure 1 shows a simple FMS scheduling problem with three workstations (W_1, W_2, W_3), three jobs (J_1, J_2, J_3) run in each workstation and each job requires two kind among four operations (O_1, O_2, O_3, O_4). Note that in this example an operation may be transferred from one job to any of others.

Table 1 shows the processing time of each operation on a workstation as a simple example of a FMS. The FMS consists of three flexible workstations.
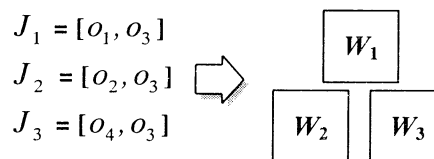
$$J_1 = [O_1, O_3]$$
$$J_2 = [O_2, O_3]$$
$$J_3 = [O_4, O_3]$$

Figure 1. Example FMS scheduling problem

Table 1. Workstation and processing time data of each operation for example FMS

Workstation	O_1	O_2	O_3	O_4
W_1	20	20	-	39
W_2	26	-	52	41
W_3	-	30	38	-
Average	23	25	45	40

Table 2. Job-related data for example of FMS problem

Job no.	Required operations	t_{ATP}	t_{DD}	c_{TP}
J_1	$O_1 \rightarrow O_3$	68	94	1
J_2	$O_2 \rightarrow O_3$	70	100	1
J_3	$O_4 \rightarrow O_3$	85	101	1

t_{ATP} : The average total processing time,

t_{DD} : The due date , c_{TP} : The total penalty cost

Collectively, these three workstations can perform four operations. No single workstation can perform all jobs but more than one station may be able to perform a given jobs. For instance, workstation W_1 can perform only operations O_1, O_2, and O_4. However, operation O_1 can be performed on W_1 and W_2 workstations. This is reflected in the different processing times for the same operation across different workstations.

Table 2 depicts relevant data pertaining to three jobs labeled J_1, J_2 and J_3 that must be scheduled in this system. Shown in the table are the operations required along with precedence requirements, average total processing time, due date, and tardiness penalty for each operation. We explain these quantities using job J_1 as an example. We assume that the batch size of each job is 1 unit. Job J_1 requires two operations O_1 and O_3 to be performed on that sequence. The average total processing time t_{ATP} or this operation is the sum of the average processing time for operations O_1 and O_3 in the system. From information in Table, we compute this quantity as 23 + 45 = 68 time units. The due date t_{DD} is actually the time specified for completion at the start of the scheduling process. For job J_1, it is 94. We assume that due dates are deterministically known a priori. Finally, the last column in the table contains information on the penalty cost per unit time associated with each job. Thus, each job, we may assess a total penalty cost c_{TP} on completion, based on the total number of time units by which an operation is tardy. For job J_1 (and all other jobs) this quantity is 1/unit time. In reality, more complex penalty cost structures could prevail. The remainder of the table is interpreted in a similar fashion.

3. Mathematical Formulation

In FMS environment, the objectives of the scheduling problem are to minimize the makespan t_M, total flow time t_F, and total tardiness penalty p_T of the project.

$$\min t_M = t_{mn} \quad (1)$$

$$\min t_F = \sum_{j=1}^{J} t_{mn}(J_j) \quad (2)$$

$$\min p_T = \sum_{j=1}^{J} \max\{t_{mn}(J_j) - t_{DD}(J_j), 0\} \times c_{TP}(J_j) \quad (3)$$

s.t.

$$t_{ik} - t_{(i-1)k} \geq p_{(i-1)k}, \forall_i \in S_i \quad (4)$$

$$t_{ik} \geq 0, i = 1,2,\cdots,m,\cdots,I, k = 1,2,\cdots,n,\cdots K$$

$$k = 1,2,\cdots,n,\cdots K, J_j \geq 0, j = 1,2,\dots J$$

where

t_{ik} : (the integer valued) finish processing time of

operation O_i on workstation W_k.

p_{ik} : the processing time of operation O_i on workstation W_k.

$A_{t_{ik}}$: the set of operations at the time t_{ik}.

$t_{mn}(J_j)$: the end processing time of last operation O_i on workstation W_k about each job J_j.

$t_{DD}(J_j)$: the due date of the job J_j (the promised delivery time of order).

$c_{TP}(J_j)$: the total penalty cost of the job J_j.

Equation (1) defines makespan that the end processing time of last operation O_i on workstation W_k. Equation (2) defines total flow time that the end processing time of last operation O_i on workstation W_k about each job J_j. Equation (3) defines the sum of tardiness costs for all orders, where the tardiness cost for a job is the multiplication of its unit tardiness cost and the absolute difference between its completion time and its processing time, given that the former is larger than the latter. Constraint (4) ensures that none of the precedence constraints are violated.

4. The Hybrid Genetic Algorithm

In Zhou and Gen's method [3], only the mutation operation was adapted because it is easy to hybrid the neighborhood search technique to produce more adapted offspring. This hybrid mutation operation provides a great chance to evolve an optimal solution. Many heuristics have been put forward and used for real-world scheduling problems. When scheduling jobs on a single machine, for example, the mean flow time is minimized by sequencing the shortest processing time t^{SP} job first, where the mean flow time is the total flow time divided by the number of jobs [4][8]. Actually, as to the FMS problems shown in Figure 1, the alternative state at each stage can be expressed by a series of integers to indicate the node or state. The FMS solution can be concisely encoded in a state permutation format by concatenating all the set states of the stages as shown in Figure 2.
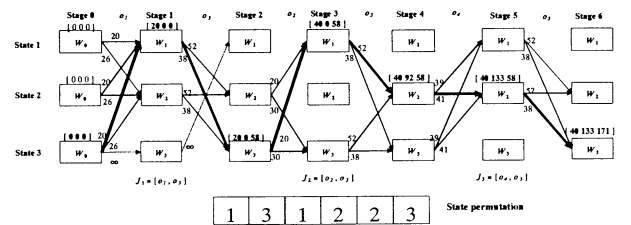

Figure2. State permutation encoding

The state permutation encoding is an one-to-one mapping for the FMS problem. It is also easy to decode and evaluate. This state permutation encoding has two valuable advantages: (1) the encoding length is only n-1 for an n-stage FMS problem, which save more computational memory while the problem scale gets larger; (2) the encoding is able to able to generate all feasible individuals in genetic operations such as crossover or mutation, which is also to raise the computation efficiency because no modification procedure is necessary for the GA operation;

$$S_3 = S_{31} \rightarrow S_{32} \rightarrow S_{33} = < J_1 \rightarrow J_2 \rightarrow J_3 >$$
$$= < (J_1,o_1)/(W_1 : 0,20),(J_1,o_3)/(W_3 : 20,58) \rightarrow$$
$$(J_2,o_2)/(W_1 : 20,40),(J_2,o_3)/(W_2 : 40,92) \rightarrow$$
$$(J_3,o_4)/(W_2 : 92,133),(J_3,o_3)/(W_3 : 133,171) >$$

4.1 Swap Mutation

The swap mutation operator [7] was used here, which simply selects two positions (jobs) at random and swaps their contents as shown in figure 3. In the example, Job sequence changes from $J_1(o_1,o_3) \rightarrow J_2(o_2,o_3) \rightarrow J_3(o_4,o_3)$ to $J_2(o_2,o_3) \rightarrow J_1(o_1,o_3) \rightarrow J_3(o_4,o_3)$ through swap mutation.
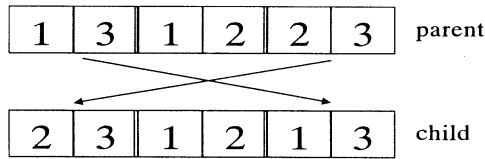

Figure 3. The swap mutation operator

4.2 Local search-based mutation (LSM)

Local search methods seek improved solutions to a problem by searching in the neighborhood of an incumbent solution [5]. The implementation of local search requires an initial incumbent solution, the definition of a neighborhood for an incumbent solution, and a method for choosing the next incumbent solution. A neighborhood of a chromosome is then defined as a set of chromosomes generated by such pair-wise interchanges. For a pair of genes, one is called pivot which is fixed for a given neighborhood and the other is selected at random as shown in figure 4. The scheme of the mutation operator is described below:
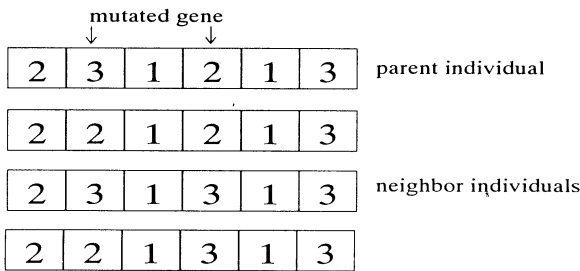

Figure 4. The incumbent chromosome and its neighborhood

4.3 result of using Hybrid genetic algorithm

As state permutation is [2 2 1 3 1 3], we can find the optimal schedule as follows:
$$S_3 = S_{33} \rightarrow S_{31} \rightarrow S_{32} = < J_3 \rightarrow J_1 \rightarrow J_2 > =$$
$$< (J_3,o_4)/(W_2 : 0,41),(J_3,o_3)/(W_2 : 41,93) \rightarrow$$
$$(J_1,o_1)/(W_1 : 0,20),(J_1,o_3)/(W_3 : 20,58) \rightarrow$$
$$(J_2,o_2)/(W_1 : 20,40),(J_2,o_3)/(W_3 : 58,96) >$$

New heuristic and its dynamic programming realization:
Makesapan t_M =96
Total flow time t_F =93+58+96=247
Total tardiness cost c_T =0+0+0=0
We could found the best optimal solution of makesapan (t_M = 96) using the hGA. Then we also found the total flow time, total tardiness cost and total transfer time.

4.4 Overall hGA approach

In this section, we show the combined procedures of hGA as follows:
Step 1. Initialization
Generate randomly an initial set of m individuals based on state permutation.
Step 2. Initial evaluation
Evaluation all individuals and calculate their fitness.
Step 3. Genetic operations
Swap mutation of jobs
Local search-based mutation of operations
Step 4. Evaluation
The fitness evaluation of offspring is implemented an objective function suggested in this paper. A best schedule for minimizing makespan, total flow time and total tardiness penalty are respectively stored in the present generation. Best schedule may be increased during optimization process of hGA.
Step 5. Termination check
If one of the individuals has lastly achieved the predefined maximum fitness, stop and return the best individual. Otherwise generate a new selection set.
Step 6. Continue with step 3.

5. Numerical Experiments

The proposed algorithms are implemented in JAVA language on PC Pentium 1400 MHz clock-pulse and 256MB RAM as operation system. The evolutionary environment for the problem was set as follows: population size was 1000, swap mutation and local search based mutation were 0.3, respectively, and maximum generation was 500. Table 3 shows the processing time of each operation on a workstation as a simple example of a FMS. The FMS consists of five flexible workstations denoted as W_1 ,..., W_5 . Collectively, these five workstations can perform five operations $o_1,...,o_5$. The required operations of each job with precedence requirements are listed in Table 4. To investigate the impact of changing between workstation, we give transfer time as the table 6.

Table 3. Workstation and processing time data of each operation for example FMS

Workstation	Operations				
	O_1	O_2	O_3	O_4	O_5
W_1	8	∞	6	∞	12
W_2	10	12	∞	9	4
W_3	∞	∞	8	10	7
W_4	∞	9	8	7	∞
W_5	7	7	10	∞	8
Average	8.33	9.33	8	8.67	7.75

Table 4. Job-related data for example of FMS problem

Job no.	Required operations	t_{ATP}	t_{DD}	c_{TP}
J_1	$O_1 \to O_2 \to O_3$	25.66	50	10
J_2	$O_1 \to O_3$	16.33	50	10
J_3	$O_1 \to O_4 \to O_5$	24.75	50	10
J_4	$O_2 \to O_4 \to O_1 \to O_5$	34.08	50	10
J_5	$O_2 \to O_5$	17.08	50	10
J_6	$O_2 \to O_4 \to O_3 \to O_5$	33.75	55	10
J_7	$O_1 \to O_5$	16.08	55	10
J_8	$O_1 \to O_4 \to O_2 \to O_3$	34.38	55	10
J_9	$O_1 \to O_4 \to O_3$	25	55	10
J_{10}	$O_2 \to O_3 \to O_5$	25.08	55	10

Table 5. The compared result with past researched methods

	t_{SAP}	DDP	Yang's GA-DDP	Proposed Algorithm
t_M	95	88	60	50
t_F	453	440	415	366

Table 6. Alternative schedules with optimal solution in FMS environment

	Alternative schedules	t_M	t_F	p_T
1	$J_6 \to J_3 \to J_4 \to J_{10} \to J_5 \to J_2 \to J_1 \to J_8 \to J_9 \to J_7$	50	366	0
2	$J_1 \to J_{10} \to J_8 \to J_5 \to J_4 \to J_3 \to J_9 \to J_2 \to J_6 \to J_7$	50	367	0
3	$J_5 \to J_7 \to J_9 \to J_4 \to J_6 \to J_8 \to J_2 \to J_{10} \to J_1 \to J_3$	50	370	0
4	$J_7 \to J_6 \to J_{10} \to J_8 \to J_4 \to J_5 \to J_3 \to J_9 \to J_1 \to J_2$	50	390	0
5	$J_1 \to J_{10} \to J_7 \to J_5 \to J_4 \to J_9 \to J_6 \to J_8 \to J_3 \to J_2$	50	396	0
6	$J_5 \to J_4 \to J_6 \to J_8 \to J_1 \to J_{10} \to J_9 \to J_2 \to J_7 \to J_3$	50	393	0
7	$J_{10} \to J_4 \to J_3 \to J_6 \to J_8 \to J_1 \to J_2 \to J_9 \to J_5 \to J_7$	50	393	0
8	$J_6 \to J_8 \to J_2 \to J_4 \to J_1 \to J_{10} \to J_3 \to J_9 \to J_5 \to J_7$	50	398	0
9	$J_5 \to J_{10} \to J_4 \to J_2 \to J_9 \to J_6 \to J_1 \to J_3 \to J_8 \to J_7$	50	400	0
10	$J_9 \to J_{10} \to J_4 \to J_6 \to J_1 \to J_8 \to J_2 \to J_3 \to J_5 \to J_7$	50	404	0
11	$J_4 \to J_6 \to J_{10} \to J_8 \to J_1 \to J_9 \to J_7 \to J_5 \to J_3 \to J_2$	50	405	0
12	$J_8 \to J_4 \to J_{10} \to J_6 \to J_7 \to J_2 \to J_5 \to J_3 \to J_9 \to J_1$	50	414	0
13	$J_4 \to J_8 \to J_9 \to J_7 \to J_6 \to J_5 \to J_1 \to J_{10} \to J_2 \to J_3$	50	415	0

Using network-based hGA which we proposed, we obtained the result (makespan $t_M = 50$ and total flow time $t_F = 366$) better than Yang's GA-based discrete dynamic programming (GA-DDP), Discrete dynamic Programs (DDP) and shortest average processing time

t_{SAP} are 60, 88 and 95 respectively as shown in Table 5.

We found fourteen kinds of alternative schedules with optimal makespan ($t_M = 50$) through network-based hGA as shown in the Table 6. Then we also found the total flow time t_F and total tardiness cost p_T. If some schedule has a problem, we can change another schedule. Alternative schedule is very important part in FMS problem.

6. Conclusion

In this paper, we integrated process planning and scheduling in a FMS environment characterized by precedence and alternative workstations with different processing times. And we formulated the FMS problem as a mathematical model, proposed the network based-hGA through local search-based mutation and swap mutation. In computational experiment, it is found that the proposed network based-hGA approach could produce best results in terms of the makespan, total flow time and total tardiness penalty. Then we could find some alternative schedules which are very important part with optimal makeapan in FMS problem. Further numerical and comparison studies are needed to explore the approach in order to provide useful guideline as to the circumstances in which the approach can be properly used.

References

[1] D.D. Bedworth and J.E. Bailey, *Integrated Production Control System*. New York: Wiley, 1987.

[2] F. Glover and M. Laguna, *"Tabu Search"*, Kluwer Academic Publishers, 1997.

[3] G. Zhou and M. Gen, "Genetic Algorithm Approach on Multi-criteria Minimum Spanning Tree Problem", *European Journal of Operation Research*, Vol.114, pp.141-152, 1999.

[4] H. Elmaraghy, V. Patel, and I. B. Abdallah, Scheduling of manufacturing systems under dual-resource constraints using genetic algorithms, *Journal of Manufacturing systems*, Vol. 19, No. 3, pp.186-201, 2000.

[5] H. Ishibuch, and T. Murata, A multiojective genetic local search algorithm and its application to flow shop scheduling, *IEEE Transactions on Systems, Man and Cybernetics*, Vol.28, No.3, pp.392-403, 1998.

[6] J. B. Yang, "GA-Based Discrete Dynamic Programming Approach for Scheduling in FMS Environment", *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetic*, Vol. 31, No. 5, pp. 824-835, 2001.

[7] M. Gen and R. Cheng, *Genetic Algorithm and Engineering Optimization*, John Wily and Sons, 2000.

[8] W. H. Clyde and S. J. Varghese, "Genetics-Based Hybrid Scheduler for Generating Static Schedules in Flexible Manufacturing Contexts", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 4, pp. 953-972, 1993.

Hybrid Genetic Algorithm with Fuzzy Goals for Optimal System Design

Masato Sasaki, Mitsuo Gen & T.Z. Dai
Dept. of Inform. & Sys. Engg.
Ashikaga Inst. of Tech.
Ashikaga 326-8558

Abstract

In this paper, a hybrid genetic algorithm based method for solving fuzzy multiple objective Knapsack problem with GUB (generalized upper bounding) structure is introduced. In this genetic algorithm, we propose the new chromosomes representation which represents the GUB structure simply and effectively at a time.

Also, by introducing the hybrid genetic algorithm(HGA) that makes use of the peculiarity of GUB structure, the proposed approach is efficient in finding solution. By the proposed approach, the solution can be searched efficiently.

1 Introduction

In general, many problems are considered to have multiple objective functions in the real world problem. Further, the goal of each objective function is considered to be fuzzy because of the Decision Maker(DM)'s subjective judgement. The fuzzy mathematical programming approach is effective to represent the real-world situations.

In some kind of large-scale optimal design problem such as optimal reliability design problem, it exists the particular structure called GUB by introducing 0-1 variable. Considering the GUB structure especially in large scale problem is important to apply the solution method.

In this paper, we propose a method for solving fuzzy multiple objective Knapsack problem with GUB structure by HGA. This approach enables the flexible optimal system design. In this GA, we propose the new chromosomes representation which represents the GUB structure simply and effectively at a time. Also, by introducing the HGA which combine the proposed heuristic algorithm that makes use of the peculiarity of GUB structure with GA, the proposed approach is efficient than the previous method in finding solution.

To demonstrate the effectiveness of the proposed approach, a Numerical Example is introduced.

2 Fuzzy Multiple Objective Knapsack Problem with GUB

2.1 fmokp/GUB Model

The fuzzy multiobjective knapsack problem with the GUB constraints(fmokp/GUB) is shown as follows:

[fmokp/GUB:]
$$\max \quad f_k(x) = \sum_{i=1}^{n} \sum_{j=1}^{n_i} c_{kij} x_{ij} \gtrsim h_k, \quad k = 1, \cdots, q \quad (1)$$

$$\text{s. t.} \quad g(x) = \sum_{i=1}^{n} \sum_{j=1}^{n_i} a_{ij} x_{ij} \lesssim b \quad (2)$$

$$\sum_{j=1}^{n_i} x_{ij} = 1, \quad i = 1, \cdots, n \quad (3)$$

$$x_{ij} = 0 \ or \ 1, \quad \forall i, j \quad (4)$$

$$a_{ij} \geq 0, \quad \forall i, j, \quad b \geq 0 \quad (5)$$

where eqs.(4) are called the GUB constraints. GUB constraints has the particular structure that each GUB constraint has only one variable with the value 1, and others 0.

$f_k(x)$ is the kth objective function, the symbol $\gtrsim$ represents fuzzy inequation. The constraints in eq.(2) are fuzzy system constraints which are given the upper and lower bound of each right hand side(RHS) constant b by the DM.

Further, c_{kij} is j-th coefficients of objective functions in subsystem i for k-th objective function, a_{ij} is the j-th coefficients of constraint in subsystem i. x_{ij} is the j-th decision variable in subsystem i. Also, n_i is the number of decision variable in subsystem i.

2.2 Membership Functions Representing the DM's Preference

We define the membership functions $\mu(f(x))$ for objective function $f_k(x)$ that represent the DM's basic

preference. Further, the membership functions for system constraints $g(x)$ as $\mu(g(x))$ as follows:

$$\mu(f_k(x)) = \begin{cases} 1 & ; \ f_k(x) > h_k \\ \dfrac{f_k(x) - z_k^-}{h_k - z_k^-} & ; \ z_k^- \leq f_k(x) \leq h_k \\ 0 & ; \ f_k(x) < z_k^- \end{cases} \quad (6)$$
$$k = 1, \cdots, q$$

$$\mu(g(x)) = \begin{cases} 1 & ; \ g(x) > b \\ \dfrac{g(x) - b + \delta}{\delta} & ; \ b - \delta \geq g(x) \geq b \\ 0 & ; \ b - \delta > g(x) \end{cases} \quad (7)$$

where h_k is the goal value for kth objective function which is between worst solution(NIS) z_k^- and positive ideal solution(PIS) z_k^*. δ is the range of the RHS constant b which is settled by the DM,

2.3 Transformation to ckp/GUB

To represent the membership functions for objective functions and constraints the fmolp/GUB is transformed to the following single-objective knapsack problem with GUB constraints by using minimum operator.

[ckp/GUB]

max $\quad z(x) = \displaystyle\sum_{k=1}^{q+1} w_k \mu_k$

s. t. $\quad \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n_i} c_{kij} x_{ij} + (h_k - z_k^-)\mu_k = z_k^-, \quad k = 1, \cdots, q$ (8)

$\displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n_i} a_{ij} x_{ij} + \delta\mu_{q+1} = b$ (9)

$\displaystyle\sum_{j=1}^{n_i} x_{ij} = 1, \quad i = 1, \cdots, n$ (10)

$x_{ij} = 0 \ or \ 1, \quad \forall i, j$ (11)

$a_{ij} \geq 0, \quad \forall i, j \quad b \geq 0$ (12)

3 Hybrid Genetic Algorithm

The proposed GA introduces the heuristic approach for finding better solution at each generation which make use of the GUB structure. The proposed genetic algorithms has the following characteristics:

- Introduce a chromosome representation which represents the GUB structure simply and effectively.

- Introduce a search algorithms which make use of the GUB structure and find solutions efficiently to GA(hybridize).

3.1 Chromosome Representation

We introduce the new chromosome representation which make use of the peculiarity of the GUB structure.

s_i is the position of variables with 1, $1 \leq s_i \leq q_i$ in subsystem i. That is, each gene $s_i, i = 1, \cdots, n$ is represented as follows:

$$s_i = j \quad \text{if} \quad x_{ij} = 1, \quad i = 1, \cdots, n, \quad (13)$$
$$1 \leq j \leq n_i$$

From this chromosome representation, the p-th chromosome V_p in a generation is represented as follows:

$$V_p = [s_{1p} \ s_{2p} \ \cdots \ s_{np}]$$

3.2 Evaluation

$$eval(V_p) = pen(V_p) * z(V_p), \quad p = 1, \cdots, pop\_size$$

where penalty function $pen(V_p)$ is expressed as follows:

$$pen(V_p) = \begin{cases} g(V_p)/b; & g(V_p) - b < 0 \\ 0 & ; otherwise, \quad p = 1, \cdots, pop\_size \end{cases} \quad (14)$$

As to the fitness function for survival, we introduce the following evaluation function based on convex fuzzy decision and evaluate each chromosome. The penalty value is assigned in proportion to the excess of the capacity of the knapsack.

3.3 Genetic Operations

The genetic operations introduced in this paper is shown below. As described above, the proposed method enables the chromosome representation which represent the GUB structure effectively, the genetic operation suitable for the problem among various operations proposed so far can be applied. In this paper, to demonstrate the effectiveness of the proposed chromosome representation, we apply the simple genetic operations.

3.4 Sorting based on the Efficient Index

Here, sort each decision variables of each GUB structure based on ranking.

Step 1 : In the same GUB constraint, gain the efficient index value for ranking the variables at each GUB constraints.

$$e_{ij} = \sum_{k=1}^{q} w_k c_{kij} \ / \ a_{ij}, \quad i = 1, \cdots, n, \ j = 1, \cdots, n_i$$

Step 2 : At each GUB constraint, rank the decision variables in descending order based on the efficient index.

$$\bar{t}_i = \ [\bar{j}_{i1} \ \bar{j}_{i2} \ \cdots \ \bar{j}_{in_i}] \qquad (15)$$
$$= \ \underset{j}{\text{indsort}}\{e_{ij} \mid j = 1, \cdots, n_i\}, \ \forall i$$

where indsort represents the sort in descending order.

Step 3 : At each GUB constraint, sort variable based on the ranking in descending order.

3.5 Overall Procedure

Step 1 : Parameter setting : Set maximum generation number max_gen, population size pop_size, crossover probability p_c and mutation probability p_m, respectively. Let generation number r = 0.

Step 2: Weights and goals setting: Set the weights w_k, $k = 1, \cdots, q$, goals of objective function h_k, $k = 1, \cdots, q$. Set the tolerable range δ for constraint b.

Step 3: Formulate the single objective nonlinear integer programming problem(ckp/GUB).

Step 4: Sort based on rank.: In each GUB constraint, sort each decision variables based on efficiency index in descending order.

Step 5 : Generate initial group : Generate initial chromosomes $\mathbf{V}_p$ with n elements at random around rank 1.

Step 6 : Gain the evaluation function value.

Step 7 : Genetic operations : Operate the following three genetic operations.
5.1 Crossover : Execute the single-point crossover.
5.2 Mutation : Execute the mutation for genes selected at random.
5.3 Selection : Select chromosomes by elitist selection.

Step 8 : Improvement solution at each chromosome (I): If the chromosome is feasible, find a chromosome which improves the objective function value from chromosomes which change gene with one rank upper and lower variable at each GUB structure. Go to Step9.

Step 9: Improvement solution at each chromosome (II): If chromosome is infeasible, find a chromosome which has the largest cost value from chromosomes which change gene with one rank upper and lower variable at each GUB structure.

Step 10 : Judge for termination: If $r < max\_gen$, then let $r = r + 1$ and go back to Step 7. Otherwise, present the chromosome with the largest evaluation value as the best solution and if the DM satisfies the solution gained, terminate the algorithm. Otherwise, set r = 1 and go to Step 2.

4 Numerical Example

As a numerical example, the Knapsack problem with two objective functions and GUB constraints is introduced.

$$\text{max} \quad f_1(x) = \sum_{i=1}^{14} \sum_{j=1}^{n_i} c_{1ij} x_{ij} \gtrsim h_1$$

$$\text{min} \quad f_2(x) = \sum_{i=1}^{14} \sum_{j=1}^{n_i} c_{2ij} x_{ij} \lesssim h_2$$

$$\text{s. t.} \quad g(x) = \sum_{i=1}^{14} \sum_{j=1}^{n_i} a_{ij} x_{ij} \lesssim 200$$

$$\sum_{i=1}^{n_i} x_{ij} = 1, \quad i = 1, \cdots, 14$$

$$x_{ij} = \quad 0 \ or \ 1, \quad \forall i, j$$

where c_{1ij} and c_{2ij} are the coefficients of objective function $f_1(x)$ and $f_2(x)$.

Let $p_c = 0.5$. $p_m = 0.6$, $pop\_size = 100$, and weights and goals $w_1 = 0.7$, $w_2 = 0.3$, $h_1 = 0.98$, $h_2 = 100.0$ and $\delta = 65.0$, respectively.

After 3rd iteration, let the DM is satisfied with the solution obtained. The reliability $R(x) = 0.964(\mu_1 = 0.97)$ and the cost $C(x) = 91.0(\mu_2 = 0.71)$, respectively. The Fitness function value $F(x) = 0.983$. Fig.

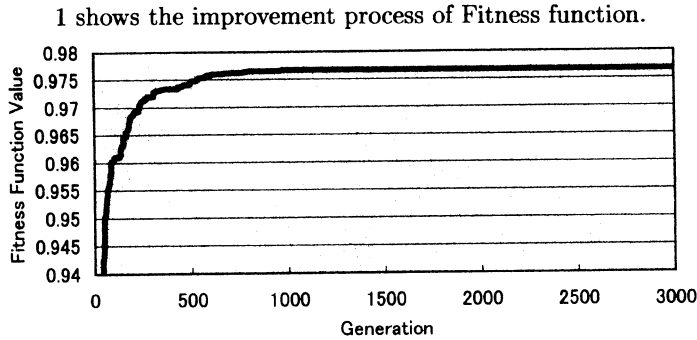1 shows the improvement process of Fitness function.

Figure 1: Improvement Process of Fitness function

In this case, the process of each membership function value at generation is shown in Fig. 2.
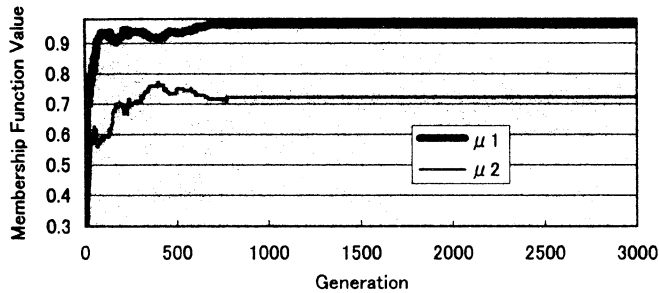

Figure 2: The Changing Process of Membership Function Values

In Fig.2, we can see that membership function values have a trade-off relationship. That is, if the membership function of $f_1(x)$ increases, the membership function of $f_2(x)$ decrease. And each membership function value converges to a certain value based on the weight w_1 and w_2 of each membership function which is given by DM.

Further, Fig. 3 represents the comparison of Parrot Optimal Solution gained by the proposed method and the method without fuzzy goals and constraint (Previous Method).
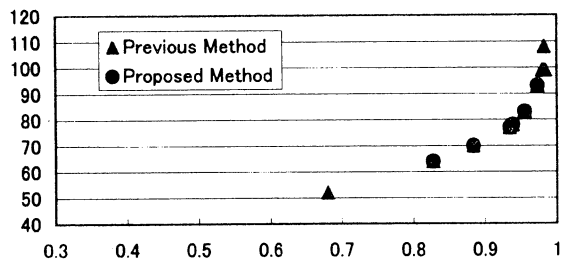

Figure 3: Comparison of Pareto Solution

In Fig. 3, we can see that the proposed method can gain the better solution of Parrot Optimal Solution. This shows the advantage of the proposed method that by introducing the fuzzy goal, the proposed method can obtain the solutions which cannot be obtained by the method because the limitation of total cost is released.

5 Conclusion

In this paper, we proposed a method for solving fuzzy multiple objective optimal system design problems with GUB structure by HGA. This approach enables the flexible optimal system design by applying extended De Novo programming. In this GA, we proposed the new chromosomes representation which represents the GUB structure simply and effectively at a time. Also, by introducing the HGA which combine the proposed heuristic algorithm that makes use of the peculiarity of GUB structure with GA, the proposed approach is efficient than the previous method in finding solution.

References

[1] Gen, M. and R. Cheng, *Genetic Algorithms & Engineering Optimization*, John Wiley & Sons Inc., 2000.

[2] Gen, M. : "Reliability Optimization by 0-1 Programming for a System with Several Failure Modes", *IEEE Trans. on Rel.*, Vol.R-24, pp.206-210, 1975.

[3] Goldberg, D. E. : *Genetic Algorithms in Search, Optimization & Machine Learning*, Addison Wesley, 1989.

[4] Hadj-Alouane, A. B. and J. C. Bean : "A Genetic Algorithm for the Multiple-Choice Integer Program", *Operations Research*, Vol.45, No.1, pp. 92-101, 1997.

[5] Y.J. Lai and C.L. Hwang : "*Fuzzy Mathematical Programming*", Springer-Verlag, 1994.

[6] Michalewicz, Z. : *Genetic Algorithms + Data Structures = Evolution Programs*, 2nd ed., Springer-Verlag, 1994.

[7] Sasaki. M. Y. Yokota & M. Gen: "A Method for Solving Fuzzy Reliability Optimization Problem by Genetic Algorithm" *J. of Japan Society for Fuzzy Theory and Systems*, Vol.7, No.5, pp. 1062-1072, 1995.

Survey on e-Manufacturing/Logistic Systems in Japan

Yinzhen Li* KwanWoo Kim** Masato Sasaki***and Mitsuo Gen***

*System Development Division, Web Technology Ltd., Tokyo 113-0021, Japan
e-mail: g-ri@webtech.fsol.fujitsu.com
**Department of Intelligent Systems, Tokyo Metropolitan Institute of Technology,
Tokyo, 190-0065, Japan, e-mail: kwkim@genlab.ashitech.ac.jp
**Department of Industrial & Information Systems, Engineering, Ashikaga Institute of Technology,
Ashikaga 326-8558, Japan, e-mail: gen@ashitech.ac.jp

Abstract

In this paper, we survey three topics on e-manufacturing and logistic systems in Japan. First, the trends of various systems report that enterprise users are interested in installing supply chain management (SCM), customer relationship management (CRM), and enterprise resource planning (ERP) in Japan. And second, we report eM-Plant that enables modeling, simulating and optimizing production systems. Finally, the used of planning and scheduling, a representation language, referred to as PSLX, for e-Manufacturing/Logistic on the internet is reported. As the related research work, we proposed an advanced (APS) is an enhancement of the cellular manufacturing paradigm, and represents the state-of-the art in the design of manufacturing systems. And we proposed a spanning tree-based genetic algorithm (st-GA) approach to find the best production/distribution design in multi stages logistic system.

Keyword: Enterprise Resource Planning (ERP), Advanced Planning and Scheduling (APS), Supply Chain Management (SCM), Planning and Scheduling Language on XML specification (PSLX).

1. Introduction

Japanese firms need to change the way they do business in order to survive in today's market. Unless firms can escape from their high-cost structure dominated by white-collar workers, reengineer themselves along global lines, and construct more flexible, low-cost information systems, their survival will be threatened. One way for solving such problems is e-manufacturing.

The development enterprise application systems are as follows: BOM (Bill of Material, 1960s), MRP (Material Requirements Planning, 1970s), MRP II (Manufacturing Resource Planning, 1980s), ERP (Enterprise Resource Planning, 1990s), and recently ERP + APS (Advanced Planning & Scheduling) included IT (Information Technology) in Supply Chain Management (SCM). Those can be summarized in the follow Figure 1:
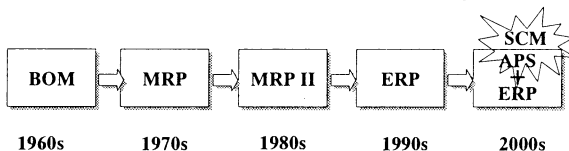

Figure 1. Evolution of planning system

The rest of this paper is organized as follows: In

Section 2 describes the enterprise application systems in Japan. And eM-Plant that enables modeling, simulating and optimizing production systems and processes will present in Section 3. In Section 4, PSLX is used for e-Manufacturing/ Logistic system on the Internet. In Section 5 describes advanced planning scheduling (APS) and logic systems in SCM. Finally some concluding remarks are given in Section 6.

2. ERP trends in Japan

The term ERP penetrated into Japanese businesses to a considerable extent. Interest was a first mainly shown by large manufactures;
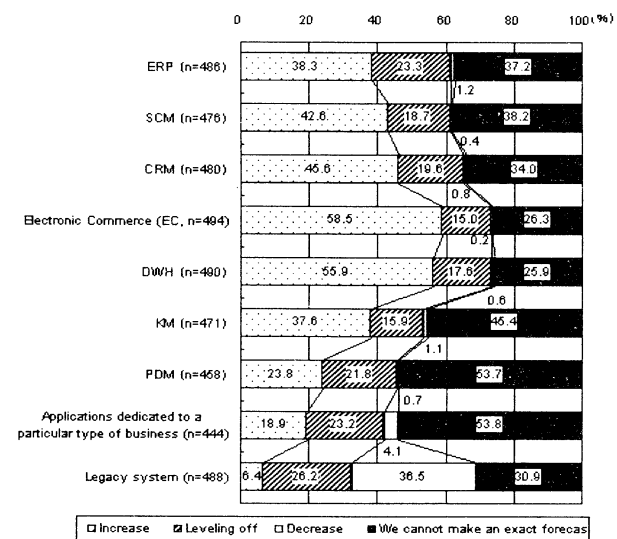

Figure 2. Increase/decrease in the numbers of application systems [3].

EC-related systems are also most important issue for Japanese enterprises to resolve at the present time. As for the firms that predicted the share of EC-related systems to intra-enterprise systems would increase over the next two years, this was the largest group of respondents at 58.5 percent (refer to Figure 2). In addition, Data Warehousing (DWH), SCM, CRM, ERP, and Knowledge Management (KM) — which are listed as keywords in the IT industry — are also mostly regarded as systems that will be actively put to use. As for legacy systems, there are few firms that are thinking of actively

increasing them, indicating that the importance of these systems will decrease in relation.

The systems that enterprise users are interested in installing are SCM, CRM, and ERP. Actually, however, the status of installed software packages or systems that can achieve these system concepts still remains relatively low. Companies that had actually installed ERP stood at only 14.8 percent as of the spring of 2001. Although the figure increased by 3.7 points compared to survey results last year, one could not say that ERP users have been dramatically increasing (refer to Figure 3).
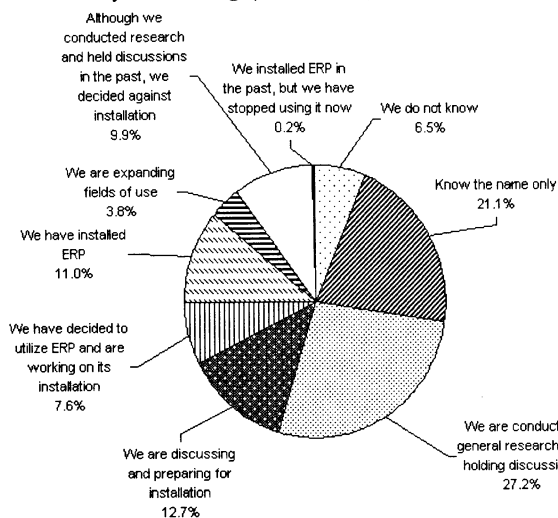
Figure 3. Status of installation and degree of ERP recognition [3].

SCM and CRM users remained only 4.8 percent and 6.9 percent as of the spring of 2001. In Figure 3, which of the following responses do you think is most similar to your company's current degree of recognition and installation status? (Only one response is allowed)

The survey showed that, except for Groupware ("We have installed and using it now" and "We are expanding its fields of use" combined came to 69.5 percent), the ratio of installation of other application systems was not very high. In terms of installed and using/expanding fields of use, EC applications and Web-EDI software accounted for 17.2 percent of the total.

Similarly, the share of responses for SCM software and CRM software accounted for only 4.8 percent and 6.9 percent, respectively, remaining at single digits. As ERP gradually penetrates the IT market, the system linkage between ERP and other enterprise applications will become more crucial.

3. e-Manufacturing solutions

eM-Plant enables modeling, simulating and optimizing production systems and processes (in figure 4). Using eM-Plant, you can optimize material flow, resource utilization and logistics for all levels of plant planning from global production facilities, through local plants, to specific lines.

eM-Plant allows simulating and optimizing production

lines in order to accommodate different order sizes and product mixes [9]. Object-oriented technology and customizable object libraries enable you to create well-structured hierarchical simulation models that include supply chain, production resources and control strategies as well as production and business processes. Extensive analysis tools, statistics and charts enable you to evaluate different manufacturing scenarios and to make fast and reliable decisions in the early stages of production planning.

Features of eM-Plant system:
● Simulation of highly complicated production systems and control strategies
● Hierarchical models of plants encompassing business and production processes
● Dedicated application object libraries for fast and efficient modeling of typical scenarios
● Graphs and charts for analysis of throughput, resources and bottlenecks
● Comprehensive analysis tools including Automatic Bottleneck Analyzer, Sankey diagrams and Gantt charts
● 3D visualization and animation
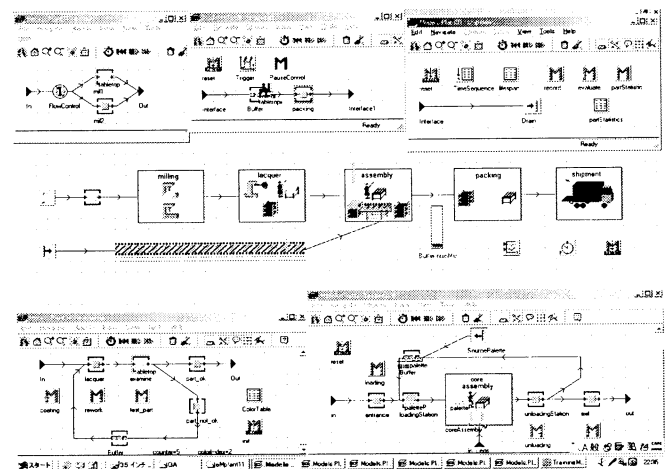● Genetic algorithms for automated optimization of system parameters

Figure 4. Plant, Line and Process Simulation

3.1   Analyzing Simulation Results

eM-Plant analysis tools allow easy interpretation of simulation results. Statistical analysis, graphs and charts display the utilization of buffers, machines and personnel. You can generate extensive statistics and charts to support dynamic analysis of performance parameters including line workload, breakdowns, idle and repair time and proprietary key performance factors. eM-Plant generates a Gantt chart of the optimized production plans that can be modified interactively.

eM-Plant is a component of the Tecnomatix eMPower line of solutions and is interoperable with other Tecnomatix software tools for developing, communicating and operating production lines and processes.

Benefits of eM-Plant system:
● Enhance productivity of existing production

facilities by 15 - 20%
- Reduce investment of planning new production facilities by 20%
- Cut inventory and throughput time by 20 - 60%
- Optimize system dimensions, including buffer sizes
- Reduce investment risks by early proof of concept
- Maximize use of manufacturing resources
- Improve line design and schedule

For additional information about why automotive, electronics, aerospace and other discrete manufacturing leaders -- such as BMW, Ford, GM, Mazda, Comau, Faurecia, Sanmina-SCI, Solectron, Nortel, Philips, Boeing, Airbus, GE Aircraft Engines, Lockheed Martin - have been turning to Tecnomatix MPM solutions, visit www.tecnomatix.com

4. PSLX: XML Specification for PPSP

Advantages of XML based representation are its openness, simplicity and scalability. Since there are many web-based technologies available to use, it is much easier to develop a new system using these new techniques rather than using conventional techniques such as EDI. XML based data exchange is becoming very popular in global manufacturing, and this will cause that connectivity is more and more convenient.

4.1 Business models using PSLX

As we briefly told an application of PSLX in visualizing the scheduling data on the Internet, there can be many series of applications depending on the PSLX standard. With respect to a supply chain management, planning and scheduling problems play a part of a key component. Therefore, collaboration and communication between scheduling software and other surrounding applications give us many valuable future business models [6].

4.2 Visualization of distributed scheduling on the Internet

The visualization of distributed schedules, which are managed in different places, is much valuable especially in a global manufacturing. If these distributed sites have their own scheduler developed by different IT vendors, their schedules cannot be visualized by a common viewer without particular adapter programs. This actually causes a huge effort of the system implementation, and both the cost and the risk of the system will be increased. Using PSLX, only these vendors prepare to develop an interface to PSLX specification, the end user needs to have one scheduling viewer such as ApstoViewer. This provides us many flexibility of visualizing distributed schedules everywhere on the Internet.

5. Related research work
5.1 Advanced planning scheduling in SCM

Advanced planning scheduling (APS) is an enhancement of the cellular manufacturing paradigm,

and represents the state-of-the art in the design of manufacturing systems. Typical advanced planning scheduling (APS) is composed of multiple workstations (or machine centers), a material handling system, and a loading-unloading station.

In this paper formulated the problem as a mathematical model and proposed a hybrid genetic algorithm approach local search-based mutation through swap mutation for integrated process planning and scheduling in a flexible manufacturing systems environment characterized by precedence and resource constraints for make-to-order and alternative workstations with different processing times [1][11]. The quality results from the evolutionary generation of order and work-in-process sequences through the swap mutation and local optimization of partial schedules in the local search-based mutation [4].
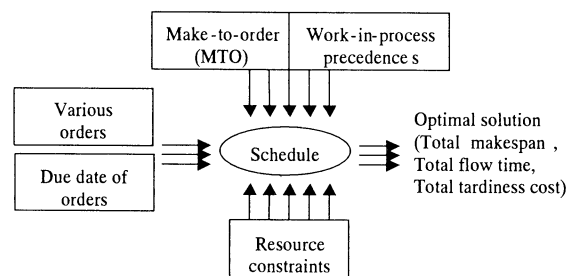

Figure 5. Basic concept of APS

In computational experiment, it is found that the proposed hGA approach could produce best results in terms of the makespan, total flow time and total tardiness penalty of order.

Figure 5 shows basic concept of APS. When customers order various products, manufacturing company produces customer products with the limited resources such as manpower, material and equipment under the appointed date of delivery. Their objective is to schedule the operations with resources constraints in manufacturing company and the precedence constraints about customers various orders.

In this Section, formulated the problem as a mathematical model and proposed a hybrid genetic algorithm approach local search-based mutation through swap mutation for integrated process planning and scheduling in a flexible manufacturing systems environment characterized by precedence and resource constraints for make-to-order and alternative workstations with different processing times.

The quality results from the evolutionary generation of order and work-in-process sequences through the swap mutation and local optimization of partial schedules in the local search-based mutation.

In computational experiment, it is found that the proposed hGA approach could produce best results in terms of the makespan, total flow time and total tardiness penalty of order.

5.2 Production/Logistic system in SCM

Logistic problem is one of the problems in operation management and operation research that requires us to make decision in several stages. In recent years, many of the developments in logistics are connected to the need of information of efficient supply chain flow to find the network strategy that can give the physical distribution flow with least cost or maximum profit [2][5][7][8]. This research is concern with logistic system design considering production/distribution planning in the view of multi-stage structure.
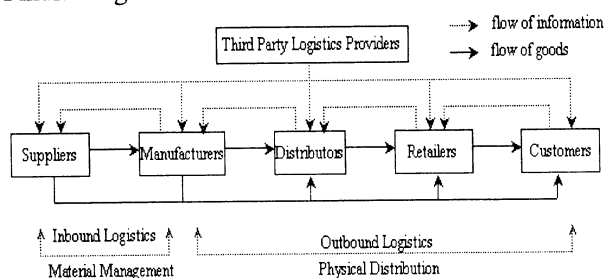
Figure 6. Supply Chain Network System

This kind of problem is usually considered as on of the supply chain network problem. Figure 6 shows the supply chain process for the logistics system which is characterized by a forward flow of goods and a backward flow of information

In this paper, the logistics system includes suppliers, plants, warehouses and customers. We consider the logistic chain network problem modeled by 0-1 mixed integer linear programming model.

The design tasks of this paper involve the choice of the facilities (plants and distribution centers) to be opened or not and the distribution network design to satisfy the demand with minimum cost. We propose the spanning tree-based Genetic Algorithm (st-GA) to design the best logistic system. In order to improve the performance of the algorithm, the fuzzy logic controller (FLC) is used to dynamically control the GA parameters [10]. The comparison with other traditional method is presented and it is shown than the proposed method is better as whole. The illustration of this problem is given in Figure 7.
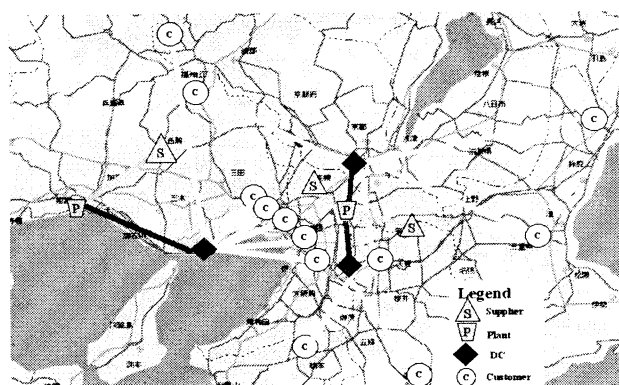

Figure 7. The illustration of the problem

In this Section, we proposed a spanning tree-based genetic algorithm (st-GA) approach to find the best production/distribution design in multi stages logistics

system. We utilize the Prufer number that is known to be an efficient way to represent various network problems. Even though the structure of the proposed methods is very simple, experimental results show that our algorithm not only can give better heuristic solution in almost all of the time but also has better performance in the sense of computational time and required memory for computation than those of matrix-based genetic algorithm (m-GA). So we believe this method will be an efficient and robust method to solve this kind of multi-stage logistic chain design problems.

6. Concluding remarks

This paper surveyed the trend of systems that enterprise users are interested in installing SCM, CRM, and ERP in Japan. It has been shown that many companies' executives in Japan have been tackling this problem to find out a way out of the present situation. To solution this problem, we proposed an advanced planning scheduling (APS) is an enhancement of the cellular manufacturing paradigm, and represents the state-of-the art in the design of manufacturing systems. And we proposed a spanning tree-based genetic algorithm (st-GA) approach to find the best production/distribution design in multi stages logistics system.

References

[1] Azevedo, A. L. and Sousa, J. P. (2000). Order planning for networked make-to-order enterprises - a case study. *J. of Operational Research Society*, Vol.51, pp.1116-1127.

[2] Brown, G. G., Graves, G. W., and Honczarenko, M. D. (2000). Design and operation of a multi-commodity production / distribution. *Management Science*, Vol.33, No.11, pp.1469-1480.

[3] ERP forum Japan (2001). Enterprise resource planning information, http://www.erp.gr.jp

[4] Gen, M. and Cheng, R. (2000). *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York.

[5] Kim, H. K. and Lee, H. Y. (2000). Production /distribution planning in SCM using simulation and optimization model, *Proceedings of SCM Korea*.

[6] PSLX Consortium Japan (2001). PSLX technical information, http://www.pslx.org

[7] Min, H. and Zhou, G. (2002), Supply Chain Modeling: Past, Present and Future, *Computers & Industrial Engineering*, in printing

[8] Syarif, A., Yun, Y. S. and Gen, M. (2002). Study on Multi-Stage Logistics Chain Network: A Spanning Tree Based Genetic Algorithm Approach, *Computers & Industrial Engineering*, in printing.

[9] Tecnomatix Technologies Ltd. (2001). eM-plant technical information, http://www.tecnomatix. com

[10] Wang, P. T., Wang G. S. and Hu Z. G. (1997). Speeding up the Search Process of Genetic Algorithm by Fuzzy Logic. *Proc. of 5th European Congress on Intelligent Techniques and Soft Computing*.

[11] Yang, J. B. (2001). GA-based discrete dynamic programming approach for scheduling in FMS environments, *IEEE Transactions on Systems, Man, and Cybernetics-part B: Cybernetics*, Vol.5, No.5, pp.824-835.

Supply chain planning in a multi-plant chain

Chiung Moon & Jongsoo Kim

Dept of Information & Industrial Engg.

Hanyang University

Ansan, 425-791, Korea

Tel:+82-31-400-5268

E-mail:cumoon@hanyang.ac.kr

Youngsu Yun

School of Automotive, Industrial & Mechanical Engg.

Daegu University

Kyungbook, 712-714, Korea

Email: joy629@hitel.net

ABSTRACT

In this paper we propose a planning model for a multi-plant chain. We formulate the problem as a mixed integer programming model with makespan criteria. It is based on the analysis of the flexible operations and sequences, resource constraints, and alternative schedules. Since the problem is *NP*-hard, a genetic algorithm is developed.

1. Introduction

Recent supply planning (SCP) methods tend to take a holistic and collaborative approach to provide global optimization. This new business model is an attempt to optimize not only plant operations but also all of the activities from a supplier to, a customer. This collaborative approach brings the idea of designing a multi-plant chain (MPC) to extend the, integration concept beyond a production site. The MPC offers new potential to increase capacity flexibility in order to timely react to market demands and customer due dates (Lutz et al., 1999).

Tan (2000), Guinet (2001), Vercellis (1999), and Kolisch (2000) have been presented some research results to solve the planning and scheduling. The major weakness of the models lies in that they consider flexible operations with a fixed sequence, infinite resource capacity, and the non-constraint operational sequence at the scheduling or planning levels.

The objective of the model is to minimize makespan while determining the operations sequences for customer orders, resource selection for each operation, schedules of the total demand of the different plants, and transportation points between plants. In order to obtain good approximate solutions, we have developed a GA based heuristic approach.

2. Model

A schedule should be determined by considering resource capacity, operations sequence, and resource selection concurrently.

The following notations are used to describe the problem throughout the whole paper:

K project mix, set of K different customer orders, *i.e.*, $K = \{1, 2, \cdots, k, \cdots, l, \cdots, K\}$.

M set of resources, $M = \{1, 2, \cdots, m, \cdots, M\}$.

Q set of lot sizes for orders in project mix, *i.e.*, $Q = \{q_1, q_2, \cdots, q_k, \cdots, q_K\}$.

G_k set of operations for order k, *i.e.*, $G_k = \left\{ g_{ki} \middle| \forall i = 1, 2, \cdots, J_k \right\}$, where g_{ki} is operation name of

the ith element and J_k is the number of operations.

p_{kim} processing time of operation i of order k on resource m.

τ_{mn} transportation time between m to n.

st_{kij} setup time from operation g_{ki} to g_{kj} on order k.

u_{kij} unit load size of order k from operation g_{ki} to g_{kj}, the number of orders in a unit load.

s_k first selected operation for order k.

A_m set of operations to be processed on resource m

C_{kim} completion time of operation i for order k on resource m

2.1 Flexible sequencing

With known unit load size u_{kij}, the number of transportation v_{kij} from operation g_{ki} to g_{kj} for q_k orders

can be calculated from the following Equation.

$$v_{kij} = \left\lceil \frac{q_k}{u_{kij}} \right\rceil \tag{1}$$

The expression $\lceil x \rceil$ represents the smallest integer greater than or equal to x. Generally, the volume per

transportation for an order between two plants is equal to its lot size. The transportation time between

plants is larger than the time within the same plant.

We use the two-commodity network flow model to represent the operations sequencing (Moon et al.,

2001). Four variables are introduced as follows:

$$y_{kij} = \begin{cases} 1, & \text{if operation } i \text{ is performed immediately after operation } j \text{ for order } k, \text{ and} \\ 0, & \text{otherwise,} \end{cases}$$

y_{kij}^p quantity of commodity p from operation g_{ki} to g_{kj} for order k.

y_{kij}^q quantity of commodity q from operation g_{ki} to g_{kj} for order k.

$$x_{kim} = \begin{cases} 1, & \text{if resource } m \text{ is selected for operation } i \text{ of order } k, \\ 0, & \text{otherwise.} \end{cases}$$

With the introduction of the variables, the transition time from operation i to j for q_k orders, c_{kij} can be

defined as follows:

$$c_{kij} = \sum_{m=1}^{M} \sum_{n=1}^{M} \left\{ q_k p_{kim} x_{kim} + v_{kij} \tau_{mn} x_{kim} x_{kjn} + st_{kij} y_{kij} \right\} \tag{2}$$

Eq. (2) is treated as the objective function for the integrated resource selection and operations

sequencing problem. The complete mixed integer programming model for the flexible operations

sequencing problem can be summarized as follows:

$$\text{Minimize} \sum_{k=1}^{K} \sum_{i=1}^{J_k} \sum_{\substack{j=1 \\ i \neq j}}^{J_k} \sum_{m=1}^{M} \sum_{n=1}^{M} \frac{1}{J_k - 1} c_{kij} \left(y_{kij}^p + y_{kij}^q \right), \tag{3}$$

$$\text{subject to} \sum_{j=1}^{J_k} y_{kij}^p - \sum_{j=1}^{J_k} y_{kji}^p = \begin{cases} J_k - 1, \text{for } i = s_k, \\ -1, \text{elsewhere,} \end{cases} \quad \forall_k, \tag{4}$$

$$\sum_{j=1}^{J_k} y_{kij}^q - \sum_{j=1}^{J_k} y_{kji}^q = \begin{cases} -(J_k - 1), \text{for } i = s_k, \\ +1, \quad \text{elsewhere,} \end{cases} \quad \forall_k, \tag{5}$$

$$\sum_{j=1}^{J_i} \left(y_{kij}^p + y_{kij}^q \right) = J_k - 1 \qquad \forall_k \text{ and } i, \tag{6}$$

$$y_{kij}^p + y_{kij}^q = (J_k - 1) y_{kij} \qquad \forall_{k, i} \text{ and } j, \tag{7}$$

$$\sum_{j=1}^{J_k} y_{kuj}^p - \sum_{j=1}^{J_k} y_{kvj}^p \geq 1, \qquad \forall_k \text{ and } (g_{ku} \rightarrow g_{kv})(g_{kv} \neq s_k), \tag{8}$$

$$y_{kij}^p \geq 0, \qquad \forall_{k, i} \text{ and } j, \tag{9}$$

$$y_{kij}^q \geq 0, \qquad \forall_{k, i} \text{ and } j, \tag{10}$$

$$\sum_{m=1}^{M} x_{kim} = 1 \qquad \forall_k \text{ and } i, \tag{11}$$

$$x_{kim} \in \{0, 1\} \qquad \forall_{k, i} \text{ and } m, \tag{12}$$

$$y_{kij} \in \{0, 1\}, \qquad \forall_{k, i} \text{ and } j \tag{13}$$

Constraints (4) and (9) are used to ensure the feasibility of the network flow of commodity p. Similarly, constraints (5) and (10) are expressed for commodity q. Constraint (6) ensures a feasible sequence. Constraint (7) explains that if $y_{kij} = 1$ the sum of commodities p and q between g_{ki} and g_{kj} is equal to $J_k - 1$. Constraint (8) explains any precedence relationship between processing operations. Constraint (11) ensures that only one resource for each operation is selected. Constraint (12) and (13) ensure the integrity of variable x_{kim}'s.

2.2 SCP model

The SCP model for the MPC involves the time allocation for operations in the selected operations sequences with the objective of minimizing makespan. To present the APS model, the following notation is introduced:

$$z_{ijm} = \begin{cases} 1, \text{if operation } i \text{ procedes operation } j \text{ on resource } m, \\ 0, \text{otherwise.} \end{cases} \quad \forall [i,j] \in A_m, m = 1,2,...,M$$

With the objective function of minimizing the makespan, the overall model can be described as follows:

$$\text{Minimize} \quad \max_{1 \leq m \leq M} \left\{ \max_{1 \leq i \leq J_k} c_{kim} \right\} \tag{14}$$

$$\text{subject to} \quad c_{ljm} - c_{kim} + W(1 - z_{ijm}) \geq p_{ljm} \tag{15}$$

$$\sum_{k=1}^{K} \sum_{i=1}^{J_k} q_k p_{kim} x_{kim} \leq \overline{W}_m \quad \forall_m, \tag{16}$$

$$c_{kim} \geq 0 \tag{17}$$

constraints (4), (5), (6), (7), (8), (9), (10), (11), (12), and (13).

where W is an arbitrarily large positive number. Constraint (15) ensures that a resource cannot be processed by more than one operation at the same time. Constraint (16) restricts the available capacity for each resource. Constraint (17) imposes a nonnegative condition.

3. GA approach

The SCP model is a NP-hard one. Any conventional optimization method is not expected to solve the problem efficiently. Therefore we decided to develop a solution methodology based on GA. GA has been successfully applied to many combinatorial optimization problems.

4. Conclusion

In this paper, a model is developed to solve the SCP problem in an MPC composed of a network of production facilities with multiple products flowing through two manufacturers. The objective is to determine the schedule of resource assignments and operations sequences of all orders so that the makespan is minimized.

To solve the problem, a GA approach is developed. To demonstrate the efficiency of the proposed GA approach on the integrated problem, numerical experiments are carried out.

Reference

Kolisch, R., 2000, Integrated assembly and fabrication for make-to-order production, *InternationalJournal of Production Economics*, 65, 289-306.

Lutz, S., Helms, S. L. and Wiendahl, H. P., 1999, Subcontracting in variable production networks, *Proceeding of the 15th International Conference on production Research*, 597-600.

Moon, C., Kim, J., Choi, G. and Seo, Y., 2002, An efficient genetic algorithm for the traveling salesman problem with precedence constraints. *European Journal of Operational Research.* 140, 606-617.

Tan Wei, 2000, Integration of process planning and scheduling-a review, *Journal of Intelligent Manufacturing*, 11, 51-63.

Vercellis, C., 1999, Multi-plant production planning in capacitated self-configuring two-stage serial systems, *European Journal of Operational Research,* 119, 451-460.

Guinet, A., 2001, Multi-site planning: A transshipment problem, *International Journal of Production Economics,* 74, 21-32.

A STUDY ON MACHINING PROCESS SIMULATION OF NC-WEDM-HS SYSTEM OF TWO TURNING COORDINATES BY MEANS OF COMPUTER

Ren Fujun   Wang Jinsong
Manufacturing Engineering Institute
Dept. of Precision Instruments & Mechanics
Tsinghua University,Beijing,100084,China
E-MAIL:renfujun@tsinghua.org.cn; renfj@public.jm.hl.cn

Abstract

In this paper, the characteristics and general laws of cutting complex curved surface by NC-WEDM-HS(Numeric Control-Wire Cut Electric Discharge Machining with High Speed) system of two turning coordinates are systematically analyzed, and the motion parameters and motion forms of NC-WEDM-HS system of two turning coordinates are also analyzed. The universal mathematical models of NC-WEDM-HS system of two turning coordinates are derived. Then, the basic idea of machining process simulation of NC-WEDM-HS system of two turning coordinates is introduced. A lot of simulating programs are programmed. Some good simulation results are obtained. Experiments on cutting complex curved surface by NC-WEDM-HS system of two turning coordinates are conducted and many complex curved surface parts are machined. The application of this simulating technology is analyzed as well. All these works provide an effective approach for us to resolve the problems of cutting complex curved surface parts by NC-WEDM-HS.

Key words: Complex curved surface, NC-WEDM-HS, Machining system of two turning coordinates, Mathematical models, Computer simulation

1 Introduction

The NC-WEDM-HS system of two turning coordinates is a definition the authors of this paper gives. This kind of machining systems is developed in order to resolve machining problem of some kinds of complex curved surface. In order to machine more parts with complex curved surface by the machining system of two turning coordinates, the basic idea of machining process simulation of NC-WEDM-HS system of two turning coordinates is presented[1].
In traditional, simulation is using model (mathematical model or physical model) to substitute practical system to do experiment. In machining process simulation of NC-WEDM-HS system of two turning coordinates, therefore, simulation is taking the digitally controlled NC-WEDM-HS system of two turning coordinates as practical system to be studied and taking mathematical models corresponding to various kinds of motion of machining process as simulating models to simulate the practical machining process of wire cutting complex curve system. The shape of complex curve to be machined on practical machining system is drown out by means of computer and the simulating chart is obtained[1].
To reach this goal, the electrode wire is taken as the generatrix in analytic geometry. Corresponding to a certain motion mode, the coordinates of every point the electrode wire passing in machining process can be calculated out and then these points are joined to form the machined curve by means of curve approaching. The chart obtained through practical data is coherence with practical part and copes with mathematical model corresponding to motion mode. Through changing the motion parameters of corresponding motion mode, different curve shape can be obtained, and the forming process of machined curve can be shown out. Further researching on the characteristics of machined complex curve, some bases can be supplied in studying the feasibility of NC-WEDM-HS system of two turning coordinates in machining complex ruled surface and the practicability of the machined curve in practical production can be obtained. Those works provide a way to resolve the machining problems of complex curved surface by the NC-WEDM-HS systems[1].

2 Establishing of the Mathematical Models of the Two Turning Coordinates Machining System

2.1 Formation of the Two turning coordinates Machining System and Its Motion Forms
The two turning coordinates WEDM system consists of NC-WEDM-HS machine made in China and a numerical

controlled rotary table, as are shown in Fig.1. In Fig.1, $\bar{X}$ stands for electrode wire's cutting motion along two directions of X-axis; $\vec{X}$ stands for positive direction of X-axis and $\bar{X}$ for negative direction of X-axis. $\ddot{A}$ expresses electrode wire's rotation around X-axis; $\vec{A}$ and $\bar{A}$ express clockwise and anti-clockwise rotation respectively. $\vec{Y}$ stands for electrode wire's cutting motion along two directions of Y-axis; $\vec{Y}$ stands for positive direction of Y-axis and $\bar{Y}$ for negative direction of Y-axis. $\ddot{C}$ expresses electrode wire's rotation around X-axis; $\vec{C}$ and $\bar{C}$ express clockwise and anti-clockwise rotation respectively.

Because of the two-axis linkage structure, including rotation and move, the two turning coordinates machining system can realize slope cutting. The other motion parameters are E, θ. E called eccentricity is the distance between electrode wire and work-piece axis; θ stands for an angle between electrode wire and work-piece axis[1].

The motion forms general equation of the two turning coordinates machining system is $\sum(\vec{A} + \vec{C} + E)$. When the general equation of the two turning coordinates machining system is expanded, there are 12 kinds of motion forms relatively independent in the two turning coordinates machining system, for example, $\ddot{A} + \ddot{C}$ etc[1].

below[1~3]:
$$\overline{OP} = \overline{OB'} + \overline{B'P} = \overline{OB'} + V\overline{\alpha} \qquad (1)$$

In formula (1), $\overline{OB'} = R\cos\phi \cdot \overline{i} + R\sin\phi \cdot \overline{j}$ is a given vector, $\overline{\alpha} = -\sin\theta_\phi\cos\phi \cdot \overline{i} - \sin\theta_\phi\sin\phi \cdot \overline{j} + \cos\theta_\phi \cdot \overline{k}$ is a given directional vector, and V is the pattern of the vector $\overline{B'P}$. So We can get

$$\overline{OP} = (R\cos\phi - V\sin\theta_\phi\cos\phi) \cdot \overline{i}$$
$$+ (R\sin\phi - V\sin\theta_\phi\sin\phi)\overline{j} + V\cos\theta_\phi \cdot \overline{k}$$

Supposing $V\cos\theta_\phi = K$, we have

$$\overline{OP} = \cos\phi(R - K\mathrm{tg}\theta_\phi) \cdot \overline{i} + \sin\phi(R - K\mathrm{tg}\theta_\phi) \cdot \overline{j} + K \cdot \overline{k}$$

And
$$\begin{cases} X = \cos\phi(R - K\mathrm{tg}\theta_\phi) \\ Y = \sin\phi(R - K\mathrm{tg}\theta_\phi) \\ Z = K \end{cases} \qquad (2)$$

Then expunge K, so the formula (2) become[2~3]:

$$\frac{X}{\cos\phi} = \frac{Y}{\sin\phi} = R - Z\mathrm{tg}\theta_\phi \qquad (3)$$

In Fig. 2, $\mathrm{tg}\theta_\phi = \dfrac{\overline{B'M}}{\overline{A'M}} = \dfrac{\overline{OB'} - \overline{O_1A'}}{\overline{A'M}} = \dfrac{R - \rho(\phi)}{h}$, then we can get the formula:

$$\theta_\phi = \mathrm{arctg}\frac{R - \rho(\phi)}{h} \qquad (4)$$

Supposing three-lead lines hexagon have n sides, we can get the polar coordinate formula of the hexagon[2~3]:

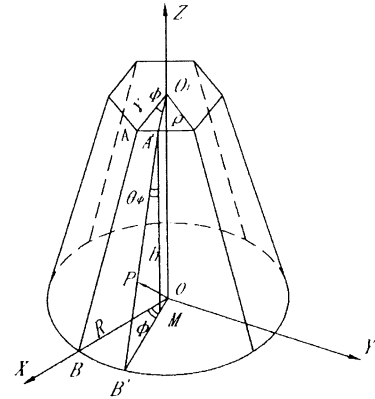$$\rho(\phi) = \frac{r\cos\dfrac{\pi}{n}}{\cos(\dfrac{\pi}{n} - \phi)} \qquad (5)$$

Fig.1 The two turning coordinates

machining system

Fig.2 Conventional diagram

three-lead lines surface

2.2 Establishing of the Typical Mathematical Models of the Two Turning Coordinates Machining System

Fig.2 is the sketch of the three-lead lines surface with a hexagon on the top and a circle at the bottom. Supposing that the radius of the circle is R, the polar coordinate formula of the hexagon is $\rho = \rho(\phi)$, and getting a point $P(X,Y,Z)$ on the surface, we can get the formula as

Using formula (5) instead of $\rho(\phi)$ in formula (4), we can get the new formula as below[2~3]:

$$\theta_\phi = \mathrm{arctg}\frac{R\cos(\dfrac{\pi}{n} - \phi) - r\cos\dfrac{\pi}{n}}{h\cos(\dfrac{\pi}{n} - \phi)} \qquad (6)$$

3 Basic Thinking about Computer Simulation of Machining Process of Two Turning Coordinates Machining System

With regard to computer simulation of machining process of NC-WEDM-HS system of two turning coordinates, it is to use the derived mathematical models as the simulation models to simulate the real process of machining work-piece in the machining system of two turning coordinates. The shape of the work-piece machined is simulated and drawn on the computer with the graphic simulation method. Thus the simulation results are obtained. In the process of simulation, the laws of machining work-piece with complex curved surface by machining system of two turning coordinates in the real machining system are discovered through analysis of the simulation result graphs. This lays a solid foundation for the study of virtual machining and the CAD, CAPP and CAM integrative technology of NC-WEDM-HS system of two turning coordinates.

According to the relevant knowledge of space analytical geometry, the work-piece surface machined is formed by the straight generatrix which electrode wire produces. Thus the coordinate of arbitrary position of electrode wire can be calculated in the process of machining work-piece. A series of positions of electrode wire are joined to form the machined surface by the approximate method of curved surface. Through coordinate transformation of practical point, the graph is obtained, which is consistent with the real machined work-piece. The graph is also in accordance with the mathematical models. By changing the relevant motion parameters, we can obtain different shapes of work-piece and find the laws about the shaping process. Then, studying further the property of machining complex curved surface, we can provide the basis for the feasibility of virtual machining of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates and for the application of the machined work-piece with curved surface in the practical production.

4 Simulation Programs Design

In the process of programming, OOP program language Visual C++ 6.0 and OpenGL1.1 are selected to be the development tool. Visual C++ 6.0 is the most popular 32 bit OOP program language under the operating system Windows 95, Windows 98 and Windows NT. Since it possesses the ideas of OOA, OOD and OOP, Visual C++ 6.0 can provide a more efficient way to develop the large

complex applied program than traditional process-oriented program language.

The predecessor of OpenGL is IRIS GL developed by SGI Corporation for its graph workstation. It is the interface of three-dimension computer graph software, which satisfies industry standard. When programming, CsplitterWnd class of VC++MFC is used to create the splitter windows. The splitter windows have two panels in the horizontal direction. Left panel is the dialog box, which is used to display, receive and transmit the parameters. So left panel is the interface which can provide input parameters and transmit data real-time. Right panel is the drawing window, responding parameters' input information. It updates the graph in terms of input information. In the process of drawing, the determined coordinate values are collected real-time and returned to CDocument. The real-time data of electrode wire's positions provide the interface for computer simulation of machining complex curved surface by NC-WEDM-HS system of two turning coordinates and the integration control.
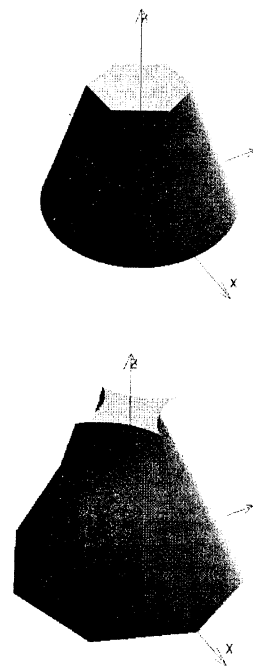

Fig. 3 Typical simulating results

In this program, the collection of data are realized by left panel; the parameters are received by CDocument through the Dialog Box. When pressing the button "OK", by calling the function UpdateALLView(), CView is told that the document data have changed. After having received the information response functions, right panel updates the data through the function OnUpdate(). Lastly, right panel uses the updated data to realize the process of

redrawing the object. Some typical simulating results are shown in Fig.3.It is necessary for us to machine some parts with complex curved surface by NC-WEDM-HS system of two turning coordinates in order to check correctness of simulating results. Some photos of parts machined are showed in Fig.4.
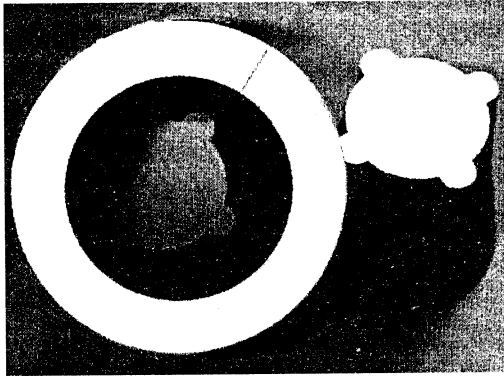

Fig.4 Some photos of parts machined

5 Conclusions

Through study on the process simulation of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates and analysis of typical simulation results, we come to the conclusion that the established mathematical models of masschining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates is correct, scientific and applicable. Using those mathematical models as simulation models, with the aid of computer, even if we don't practically machine the work-pieces, we can observe the process of machining work-piece with complex curved surface by NC-WEDM-HS

system of two turning coordinates and the surface shape of machined work-piece. If the values of the parameters are adjusted, the shape characteristics can be changed. The shape of machined work-piece can be changed by adjusting the values of a certain parameter, so the ideal

machining parameters of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates can be obtained by computer simulation. Using the ideal machining parameters to adjust the machining system, the required shape can be machined or new methods and laws can be found.

Thus the technology of computer simulation can not only examine the correctness of mathematical models of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates but also provide the basis for possibility and the feasibility. The relevant experiment can conversely prove the correctness of simulation results. Owing to save a great deal of time, we can increase the productivity and improve the machining quality of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates. At the same time, this lays a solid foundation for the study of virtual machining and the CAD, CAPP and CAM integrative technology of machining work-piece with complex curved surface by NC-WEDM-HS system of two turning coordinates.

Acknowledgements

References

[1]Zhou Zhenggan, LIU Jinchun. Research on machining multidimensional complex straight burr curved surface by WEDM-HS with three-axis linkage (in Chinese).Electro-machining (4):PP.26-32. 1994

[2]The mathematics department of FuDan university. Curve & Curved Surface (in Chinese). Science Pub. .1982

[3]The mathematics department of NanKai university. An introduction to analytical geometry of space (in Chinese).People's education Pub. 1984

Extraction of the Quantitative and Image Information from the Flame Images of Steam Boilers of the Steam Power Generation

Hyeon Bae, Dong Jae Park[*], Hang Bae Ahn, Byong-Hee Jun, Sungshin Kim, Jong-Il Bae[**], and Man Hyung Lee[***]

School of Electrical and Computer Eng., Pusan National University, Busan, Korea
[*]Ulsan Office, Korea Plant Service & Engineering Co., Ltd.
[**]Div. of Elect. Control and Instrumentation Eng., Pukyong Nat. Univ., Busan, Korea
[***]School of Mechanical Eng., Pusan National University, Busan, Korea

Pusan National University 30 Changjeon-dong, Keumjeong-ku, Busan 609-735, Korea
E-mail: baehyeon@pusan.ac.kr, eastlive@korea.com, ddasmi@pusan.ac.kr, bhjun@pusan.ac.kr,
sskim@pusan.ac.kr, jibae@pknu.ac.kr, mahlee@pusan.ac.kr

Abstract – Several equipments for flame detection are employed in power generation that use UV, IR, VL, different pressure, flame rod, and so on. But these flame detectors have some problems to detect flame with good performance. So in this paper, we apply different techniques for the flame detection. Image processing techniques are broadly applied in industrial fields. Because process speed is much improved. In this paper, the image information is employed for obtaining fire images. And then these images are preprocessed for the input values of neural network model. We can test and evaluate the approach that uses image information for the flame detection of burners. If this technique is implemented in physical plant, economical and effective operation could be achieved.

Keywords: Flame detection, neural networks

1. Introduction

A flame detector is located in the burner that is to detect the burning flame and then to generate the electric signal to the automatic burner controller. The flame detector plays an important role in maintaining the safety of the boiler during starting up/shutting down or working on that is the first goal of the burner automatic controller [1]. And the flame detector as the final safety device is connected with the fuel valve in the normal operating condition avoiding the explosion of boilers. If the fuel is supplied into the boiler combustor in the condition that lose the flame, then boilers could be exploded. At starting burners up, the flame detector sends the flame detection signal to the burner automatic controller and the signal is applied to control the igniter, sub-combustor burner, main combustor burner, operating lamp, and so on [2].

The traditional boiler flame detector has the disadvantages as follows [3]:

- It is difficult to configure the detector for the accurate detection at installing initially.
- Inspection of the operating status is unreliable, because the external measuring instrument just detects the intensity of the fire flame.
- Incorrect action can occur because the detector could not distinguish between self-flame and opposite flame.
- The maintenance of the old devices is very difficult

For shooting these disadvantages, the new flame detection method is applied that uses the captured image information and neural networks in this paper.

2. Flame Detector
2.1 Flame Detector of Power Generation

The flame detector can be divided into two types largely that are emission detection method and visible ray detection method. The emission detection method detects the flame using the distinctive quality of the electron radiant wave and flickering phenomenon that are kinds of flame features, and the visible rays detection method detects the flame using visible rays and flickering phenomenon. There are the other methods to detect the flame as flame rod method, differential pressure method, and so on [4].

Table 1. Principle of flame detector.

Detection ray	Principle of flame detection	Detection device
Ultraviolet rays (UV)	Use Photoelectric effect for acting the cathode glow electric discharge	Phototube
	Use photo electromotive effect for acting the silicon photo diode	Photo diode
Visible ray (Cd-S Cell)	Make reaction of the photo electromotive effect in the visible ray area and use the flame flicker together	Photo diode Photo transistor
Infrared ray (IR)	Use photo conductive effect when it get the light, its conductivity is increased	Lead sulfate
	Make reaction photo electromotive effect in the infrared ray area	Photo diode
	Use photo electromotive effect and flame flicker effect	Lead sulfate Photo diode
Others	Use the sudden change between initial differential pressure of burner gun and differential pressure of burning point at the initially burning	Differential pressure method
	Detect the variation of electric conductivity by rod that is caused by the production of gas ion in the flame	Flame rod

3. Neural Networks
3.1. Neural Network Model

Artificial neural network called as neural network

generally was begun from awareness, which the computation way of person's brain is quite different with the calculation method of traditional digital computer.

The brain is very complicated and non-linear, and it is parallel operator. And it has ability that organizes neurons and calculates the specific arithmetic problems fast than the digital computer through the neurons. The artificial neuron is a kind of the model that has the direct analog component for the actual neuron component.

Input signals are expressed as x_0, x_1, ..., x_n in the network diagram as shown in Fig. 4. Input signals are modified by weights [5].
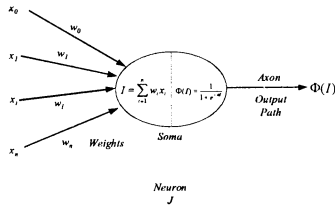
Fig. 1. Diagram of a neuron.

3.2. Image Preprocessing

Each color appears by basic spectrum factors of red, green, and blue in RGB model. The gray color is spread along the line that connects two white point from black color in this model, and the color is on the cube space or the inside point that is defined as the vector that is extended from the original point. In this paper, original RGB images are applied as the converted images to the gray scale.

$$Grey = \frac{R+G+B}{3} \quad (1)$$
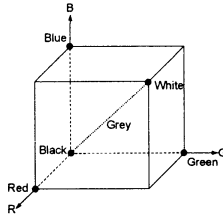
Fig. 2. The cubic of RGB colors

The values to the axes values of each image through the projection are employed only in this paper. In the projection of the binary image, the horizontal and vertical projections are achieved with adding the each pixel vertically and horizontally, respectively. In the projection of the binary image, $V[j]$ projection for row values and $H[i]$ projection for the column values are processed by following equations.

$$H[i] = \sum_{j=1}^{m} B[i,j]$$
$$V[j] = \sum_{i=1}^{n} B[i,j] \quad (2)$$

From above equations, the general projection to the specific axis can be defined.

$$A = \sum_{j=1}^{m} V[j] = \sum_{i=1}^{n} H[i] \quad (3)$$

$$\bar{y} = \frac{\sum_{i=1}^{n} i H[i]}{A},$$
$$\bar{x} = \frac{\sum_{j=1}^{m} j V[j]}{A} \quad (4)$$

4. Flame Detection using Neural Network
4.1 Experimental Site and Equipments

Ultraviolet rays phototube detector, brand Fireye 45UV5 developed by E.C.A. Co. (electronic corporation of America: USA) that is a basic model of the flame detector in this experiment is applied broadly in the heavy oil boiler.

The experimented generation boiler consists of 18 numbers of burners (6×3), it is developed by Babcock Co. and amount of the maximum generating steam is 700 $ton/hour$. Ulsan steam power generation that is experimental site can generate 200MW output and selects the above boiler. The actual experiment is accomplished with the third generator of Ulsan steam generation. The data are collected at the normal operating time and starting time.

Employed main equipments in this experiment are shown in the Table 2.

Table 2. Equipments of practical test.

No.	Name	Specification	Maker
1	Memory Recorder	32ch Hioki8826	
2	Amp Module	Fireye 25SU3	E.C.A
3	Flame Detector	Fireye 45UV5	E.C.A
4	Digital Camcorder	DCR-PC5	Sony
7	Visible Ray Cut off Filter	ND-2, ND-4	Kenko
9	Adapt for filter	30→37mm, 37→52mm	
11	Optical Filter	Red, Green, Dark Black	

The main goal of this experiment is to distinguish the existence or nonexistence of the flame of boiler burner using collected video signal data and neural network that are based on the visual judgment by human's experience without depending on the optical characteristic. The purpose of this experiment is to propose the new type of flame detection that has the basic function of flame detector and removes disadvantages of the traditional flame described in the research background.

(a) Traditional method of the flame detection.
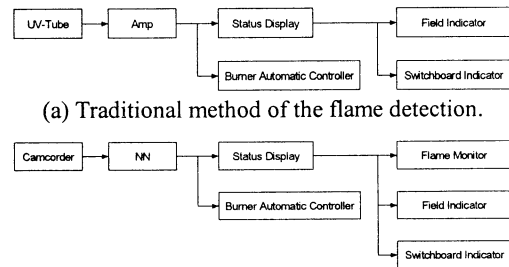
(b) Proposed method of the flame detection.
Fig. 4. Comparison of traditional and new method.

The digital camcorder at the time of burner on or burner off films flame images separately. The image date were obtained under the conditions that the exposure of the digital camcorder was manually fixed at +2 point (exposure 0 standard) by standard, one ND-2 filter and two ND-4 filters were employed together, and then the

transmissivity of the visible rays was configured at 3.125%.

4.2. Image Preprocessing

The burner on/off status was filmed by a camcorder in condition such as Fig. 4 in this paper.

(a) Burner-off (case 1)   (b) Burner-off (case 2)

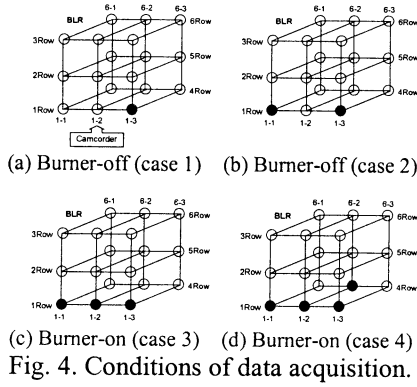(c) Burner-on (case 3)   (d) Burner-on (case 4)

Fig. 4. Conditions of data acquisition.

Fig. 5 shows images of the burner on/off status filmed in the different burner on/off condition of other neighbor burners. Figure (a) and (b) are filmed under the various on/off conditions of adjacent burners and the self-burner off. On the other hand, figure (c) and (d) are taken in the condition that self-burner is turned on.

(a) Case 1   (b) Case 2

(c) Case 3   (d) Case 4
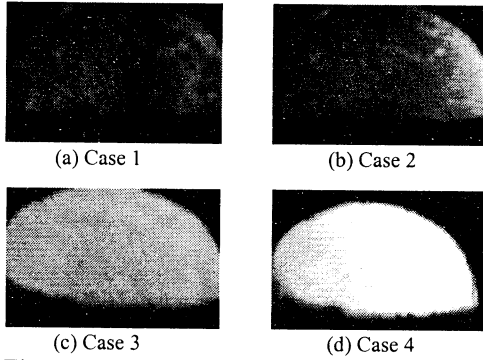
Fig. 5. Images in the cases of burner status.

In filmed record images, 10 snap images were captured according to each condition. Even though the original images could be processed directly but the captured images were converted to gray scale. Because we thought that the image classification according to the shade of light is more useful for effective judgment in this paper. Fig. 6 shows the converted images from the original image to the gray scale image.

(a) Gray image of case 1   (b) Gray image of case 2

(c) Gray image of case 3   (d) Gray image of case 4
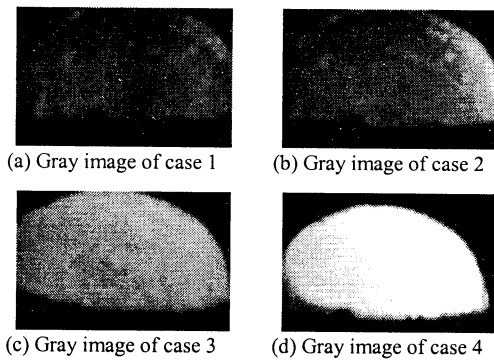
Fig. 6. Image conversion of captured images.

In this paper, we wished to use projection of images to x and y-axis. In other words, the image information is reconstituted by mapping the gray scale images. In the case of such recomposed data, the data are expressed as one-dimensional data that has definitely different values according to shade. Fig. 7 shows the results that are obtained by the projection of the captured 10 numbers of images from the filmed images under the burner off condition. The figure (a) is the result that is obtained by projecting 10 numbers of images to y-axis and the figure (b) displays the result that is achieved by mapping the images to x-axis. Fig. 8 shows the result graphs of the burner on condition images that are projected to each axis as above results.

(a) y axis projection of case 1   (b) x axis projection of case 1

(c) y axis projection of case 2   (d) x axis projection of case 2
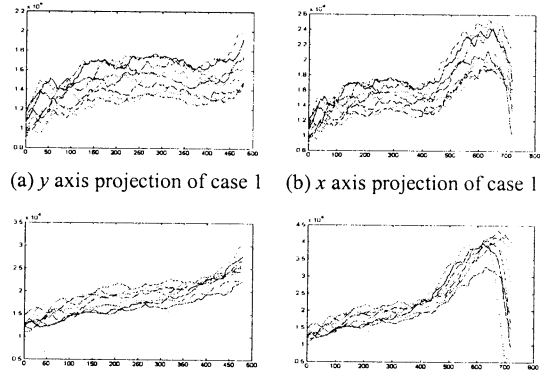
Fig. 7. Projection images of burner off conditions.

(a) y axis projection of case 3   (b) x axis projection of case 3

(c) y axis projection of case 4   (d) x axis projection of case 4
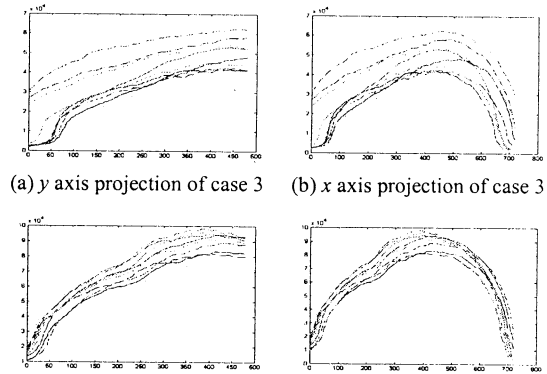
Fig. 8. Projection images of burner on conditions.

Fig. 9 is sketched by calculating each average of each 10-projection image that are acquired in two operating conditions.

(a) Average of y-axis projection (b) Average of x-axis projection
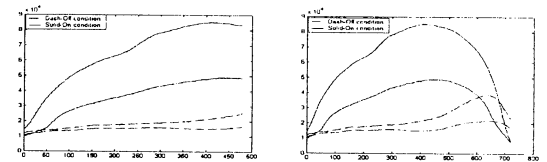
Fig. 9. Mean value graphs mapped into x and y-axis.

4.3 Fire Flame Detection using Neural Networks

Two cases data are used for training data that are divided by the condition of burner on/off and adjacent burners in this experiment. Namely the total 20 captured images corresponding to two cases of burner on

condition are composed.

Also, for the off condition, 20 images of each two cases are generated similarly. Among the prepared data, 14 numbers of data sets are used for training and 6 numbers of data sets are applied for testing. According to results that map images of each gray scale to x and y-axis, 720 samples and 480 samples generated to x-axis and y-axis, respectively. Therefore, dimension of whole input data becomes 1200×28. Output data are composed by the binary value that is used usually in neural networks as shown in Table 3.

Table 3. Structure of the target data for neural network.

Output		Target output y								
Target variable		y_1	y_2	...	y_{13}	y_{14}	y_{15}	y_{16}	...	y_{27}	y_{28}
Target Value	On case	1	1	1	1	1	0	0	0	0	0
	Off case	0	0	0	0	0	1	1	1	1	1

4.4 Results of the Flame Detection

Table 4 shows the learning results of input data. It consists of the training results and error according to target values of each case. As shown in the results, the most of trained results has over 0.9 values to target value 1. This means that it is easy to classify pattern of data because learning performance is superior.

Table 4. Results of training of the neural network.

Condition	Case	No	Target		Result		Error	
			On	Off	On	Off	On	Off
Burner On	Case 1	1	1	0	0.8667	0.2114	0.1333	-0.2114
		2	1	0	0.9095	0.0941	0.0905	-0.0941
		3	1	0	0.7818	-0.1042	0.2182	0.1042
		4	1	0	0.8891	0.1330	0.1109	-0.1330
		5	1	0	0.9139	0.4121	0.0861	-0.4121
		6	1	0	0.9515	0.3770	0.0486	-0.3770
		7	1	0	0.9412	0.4682	0.0588	-0.4682
	Case 2	8	1	0	0.9447	0.1659	0.0553	-0.1659
		9	1	0	0.9494	-0.1126	0.0506	0.1126
		10	1	0	0.9504	-0.0405	0.0496	0.0405
		11	1	0	0.9437	-0.2438	0.0563	0.2438
		12	1	0	0.9513	-0.1276	0.0487	0.1276
		13	1	0	0.9480	-0.0505	0.0520	0.0505
		14	1	0	0.9487	0.0462	0.0513	-0.0462
Burner Off	Case 1	15	0	1	-0.0433	0.9319	0.0433	0.0681
		16	0	1	0.0001	0.9237	-0.0001	0.0763
		17	0	1	-0.0227	0.9291	0.0227	0.0709
		18	0	1	0.0814	0.9249	-0.0814	0.0751
		19	0	1	0.0582	0.9336	-0.0582	0.0664
		20	0	1	0.0764	0.9298	-0.0764	0.0702
		21	0	1	0.0940	0.9205	-0.0940	0.0794
	Case 2	22	0	1	0.0572	0.8948	-0.0572	0.1052
		23	0	1	-0.0966	0.9425	0.0966	0.0575
		24	0	1	-0.0225	0.9138	0.0225	0.0862
		25	0	1	-0.0339	0.9258	0.0339	0.0742
		26	0	1	-0.0569	0.9300	0.0569	0.0700
		27	0	1	-0.0540	0.9236	0.0540	0.0764
		28	0	1	0.0187	0.8845	-0.0187	0.1155

Three numbers of test data sets of each case are tested by the trained neural network model and the performance of the model is validated whether classify pattern of image properly. As shown in Table 5 that contains the results of data for burner on/off condition, we can know that classification performance is superior. Most of the results show over 0.9 comparing

with the target value 1. This means that the model can distinguish on/off case certainly by image values. And as whole result, we can confirm that performance is superior regardless of burner on/off condition.

Table 5. Results of testing of the neural network.

Condition	Case	No	Target		Result		Error	
			On	Off	On	Off	On	Off
Burner On	Case 1	1	1	0	0.9397	0.2982	0.060277	-0.29817
		2	1	0	0.88022	0.39554	0.11978	-0.39554
		3	1	0	0.89106	0.17851	0.10894	-0.17851
	Case 2	4	1	0	0.93133	-0.33225	0.068674	0.33225
		5	1	0	0.9389	0.004659	0.061101	-0.004659
		6	1	0	0.95044	-0.36445	0.049556	0.36445
Burner Off	Case 1	7	0	1	0.038773	0.93542	-0.038773	0.064578
		8	0	1	0.022955	0.92581	-0.022955	0.074188
		9	0	1	-0.002395	0.93341	0.0023956	0.066592
	Case 2	10	0	1	-0.063831	0.92457	0.063831	0.07543
		11	0	1	0.006128	0.90827	-0.006128	0.091733
		12	0	1	-0.10378	0.93876	0.10378	0.061236

5. Conclusion

Flame detection devices of power plant, which is used present, have many disadvantages, and are depending much on optical judgment by actuality driver's experience. In this paper, the flame detection is attempted using similar information with actual person's observation on eye based on the optical image information. Filmed images are sorted according to operating condition and converted to gray scale. New image information is acquired by mapping the image to each axis. Because reconstructed data according to burner on/off condition show the definitely difference, when these data are used for input of the neural network model, the neural network model offers superior recognition performance.

References

[1] Kyung-Hae Cho, *Handbook of Steam Power Generation*. Gumi technology, pp. 357~364, 1991.

[2] "Guide book of basic operation I," Korea Plant Service & Engineering Co., Ltd., Ulsan Office, Technical Report, pp. 100~103, 1990.

[3] *Localization of burner flame detector*. Nambuk Eng. Co., Ltd., 2001.

[4] Jack A. Bryant, "Update your flame-monitoring know-how," *Power Engineering*, vol. 124, no. 8, pp. 1~6, August 1980.

[5] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*. NJ: Prentice Hall, pp. 199~253, 1997.

A Novel Method for Compression of Image Sequences Based on

Nonlinear Dimensionality Reduction

Jun Wang, Changshui Zhang
State Key Laboratory of Intelligent Technology and Systems
Department of Automation, Tsinghua University, Beijing (100084), China

Abstract

The compression of image sequence is an important problem in the field of information storage and transmission. There are many classical compression techniques, which have been used in many real-world applications. In this paper, we proposed a novel method for compression of image sequences applying a newly developed Nonlinear Dimensionality Reduction (NDR) technique, Locally Linear Embedding (LLE). As a kind of dimensionality reduction method, LLE can project source data to a much low-dimensional output space and provided the intrinsic manifold, which can be regarded as the compress step for data. We construct a nonlinear regression process to implement the decompose step and reconstruct the whole sequence based on several kernel frames. The experiments validate the efficiency of our idea.

1 Introduction

Image sequence compression is an important problem in the field of data storage and transmission. It can reduce storage requirements and make it possible to transmit the data in a low-BPS (bit per second) network. In the area of pattern analysis and image process, classical linear dimensionality reduction method, such as Principal Component Analysis (PCA) [1], Independent Component Analysis (ICA) [2,3] and Linear Discriminate Analysis (LDA) [4], can be used as compression techniques. But all these linear methods can not perform well in image sequence compression. First, the differences between frames are always nonlinear, which can not be presented well by linear space method. Second, linear dimensionality reduction method can not reveal the intrinsic manifold of the data.

Locally Linear Embedding, as a kind of Nonlinear Dimensionality Reduction method, was proposed by Sam Roweis and Lawrence Saul recently [5, 9]. LLE maps the high-dimensional data to a single global coordinate system in a manner that preserves the neighboring relationships. The method has been successfully applied to reduce the dimensionalities of artificial data and real-world data, such as "Swiss roll" data and face images. Because the intrinsic dimension and structure of the data set is always very low, the projected data of a given set distributed in a very low-dimensional embedded space. Some work in which the true manifold structure of data is revealed has also been conducted [6, 7]. LLE, however, lacks the mapping between the source data and output data. There is no doubt that LLE is a potential data compression technique.

In our work, LLE was applied to reduce the dimensionality of an image sequence set. We reveal the manifold of an image sequence and consider how to use LLE to compress the image sequence. First, we proposed a nonlinear interpolation method to reconstruct the whole sequence from several frames. Second, for better performance of reconstruction, we present the optimization methods to improve the efficiency of our method.

In this paper, the second part contributes the analysis of dimensionality reduction of image sequence by LLE. The third part describes a nonlinear method to reconstruct the whole sequence from several frames.

The fourth part does the optimization to improve the performance of the method. The final part draws some conclusions and discusses.

2 Dimensionality Reduction of Image Sequence by LLE

In this step, we apply LLE to reduce dimensionality of a given set, which is composed with image sequence. The first thing is to determine the intrinsic dimensionality of the set. Another NDR method, ISOMAP, can reveal the intrinsic dimensionality [8]. More details of the choice of embedding dimensionality can be found in Reference [13]. Based on experimental knowledge, we make the embedded space be three-dimensional space. Figure1 shows several frames selected from an image sequence. The result of dimensionality reduction is shown in Figure2.

Figure1 Ten frames selected from a image sequence.

In Figure2, the projections maintain the order of the image sequence. A result of dimensionality reduction of a complicated image sequence is exhibited in Figure3. In this figure, some source images and their corresponding projected points are shown.
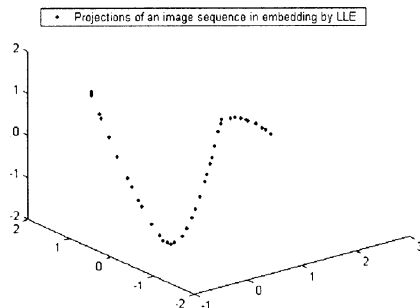
Figure2 Projections of the images in a sequence by LLE.
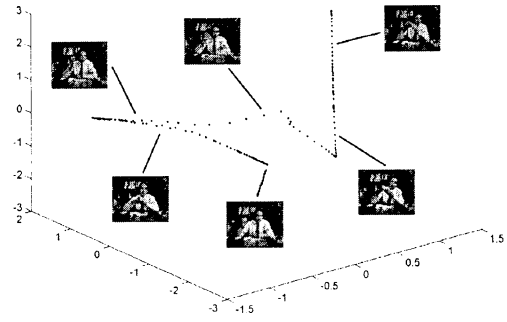
Figure3 A result of dimensionality reduction of a complicated image sequence by LLE.

3 Compression of Image Sequence by LLE

The experimental results above give us a hint that LLE can implement high-ratio compression for image sequence. However, we should construct the decompression process. We consider that methods of statistical learning theory, such as Support Vector Regression, can construct a mapping between two spaces in small set cases.

Suppose the images consist of Y_i, $Y_i \in R^N$ and the projection of Y_i by LLE algorithm is X_i, $X_i \in R^d$ (In our experiments, D is 4096, and d is 3.). We give the construction from X_i to Y_i as follows.

$$Y = F(X) \tag{1}$$

By SVR, we define the form of function $F(\cdot)$

$$Y = F(X) = \sum_{i=1}^{l} \alpha_i k(X, X_i) + b \tag{2}$$

In Equation2, X_i is the training sample and l is the sample number. The kernel function $k(X, X_i)$ decides the property of $F(\cdot)$. The parameters are a_i, b, ($i = 1,2..l$). Applying SVR, we get the definite form of the function $F(\cdot)$ on small set [10, 11, and 12]. The construction of the image sequence can be implemented by using the function and the preserved low-dimensional points.

During the process of compression, we apply LLE to the images and achieve the embedding points. Some frames and all the

low-dimensional data are preserved. For the process of decompression, the saved frames and their corresponding embedding points are used to construct mapping between embedded space and source space. The whole image sequence can be reconstructed by the mapping and low-dimensional points. How to select the key frames to make the mapping more precise is another problem, about which we give some solutions in next section.

4 Selecting Key Frames for Mapping

How to select key frames to learn the mapping better? This is another problem which should be considered. In fact, it relates to the intrinsic problem about selecting support vector in SVR. In our work, we avoid the difficult problem about SVR, but propose two solutions for our specific problem. The first solution is based on the following idea. We assume that the image sequences distribute along a high-dimensional curve. The key frames lies in the segment where the curve distorts sharply. So the similarities between neighboring frames are calculated and analyzed, then more images are sampled where the differences between frames are large. The second solution derives from the idea of combination optimization. Combinations of key frames are picket out randomly, among which the combinations with superior performance are chosen and preserved. The peak value of signal/noise ratio is used as the objective function to evaluate the performance. An optimization

algorithm, for example, genetic algorithm is applied to obtain a better combination of key frames.

Figure 1 shows a sequence with 79 frames. We only save 8 frames for implement our algorithm. Some frames of the reconstruction image sequence are shown in Figure 4.
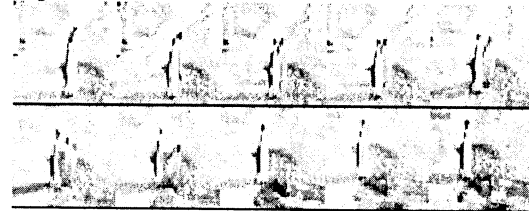
Figure4 Some frame of the reconstructed image sequence.

Peak Signal-to-Noise Ratio (PSNR) values are used to evaluate the performance, which is defined as:

$$\delta = 10\log\frac{255^2}{\sigma^2} \qquad (3)$$

$$\sigma^2 = \frac{1}{D}\sum_{i=1}^{D}(\tilde{Y}_i - Y_i)^2 \qquad (4)$$

In Equation4, $\tilde{Y}$ denote the reconstructed image, and Y_i is the corresponding source image. In our experiments, we compare our method with the MPEG4 method, Table1 show the performances of our method, named as NDR method, and several standard video algorithms. The source image sequence includes 79 frames and the image is 64×64 with 256 gray levels. The storage of source images is about 400k bites.

Table 1 Performance Comparison of Several Compression Algorithms

Compression method	Storage	Mean PSNR
Microsoft Video 1	49K	46.8 DB
Microsoft Video 1	80K	62.7 DB
Intel Video R3.2	166K	84.5 DB
MPEG4 Fast-Motion	34 K	73.5 DB
NDR Method	40K	62.4DB

In fact, the Mean PSNR can not be regarded as the only criteria to evaluate the quality of a video. Decompression of our method is based on kernel learning, so the reconstructed frames far away the kernels is not as precise as those near kernels. Thus one of the drawbacks of our method is that the reconstructed sequence is not as smooth as that of MPEG4 Fast-Motion and Microsoft Video 1 with equivalent

compression ratio.

For complicated sequence, such as that shown in Figure 3, we should segment the sequence before compression. How to make the segmentation reasonable is another issue should be addressed in our future work.

5 Conclusions

As a kind of Nonlinear Dimensionality Reduction techniques, Locally Linear Embedding can reveal the underlying manifold of a given data set and project the high-dimensional data to a very low space. In this paper, we propose a novel method of image sequence compression applying LLE. Our method is proved to be efficient by experiments. However, there are open problems, for example, the computational efficiency make it difficult for real-time applications. Furthermore the performance of our method does not achieve as high as that of classical video compression techniques, such as MPEG4. It, however, has the potential to solve some problems in video processing. First, it has the potential to be used for nonlinear prediction, which is our ongoing research. Second, our method can be used to reconstruction high-density image sequence, which gives us the method to analyze complicated motion of objects with high speed.

References

[1] M.Turk, "A. Pentland, Eigenfaces for Recognition", Journal of Cognitive Neuroscience, 1991, (3) 71-86.
[2] Marian Stewart Bartlett, Terrence J. Sejnowski, "Independent components of face images: A representation for face recognition", Proceedings of the 4th Annual Jount Symposium on Neural Computation, Pasadena, CA, May 17, 1997
[3] Marian Stewart Bartlett, "Face image analysis by unsupervised learning and redundancy reduction", Ph.D. Thesis at University of California, San Diego (1998)
[4] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specfic linear projection", IEEE Trans. on PAMI, Vol. 19, No. 7, 711-720, 1997.

[5] Sam Roweis, Lawrence Saul, "Nonlinear dimensionality reduction by locally linear embedding", Science, v.290 no.5500, Dec.22, 2000. pp. 2323--2326.
[6] Lawrence K. Saul, Sam T. Roweis, "An introduction to locally linear embedding", http://www.gatsby.ecl.ac.uk/~roweis/lle/papers/lleintro.pdf
[7] Alice Zheng, "Deconstructing motion", EECS Department U. C. Berkeley December 15, (2000), http://www.cs.berkeley.edu/~alicez/lle_walking_print.ps
[8] Joshua B. Tenenbaum, Vin de Silva, John C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction", Science, v.290 no. 5500, Dec.22, 2000. pp.2319—2322
[9] David Gering, "Linear and nonlinear data dimensionality reduction", April 17, (2002), http://www.ai.mit.edu/people/gering/areaexam/
[10] Steve Gunn, "Support vector machines for classification and regression", ISIS Technical Report ISIS-1-98, Image Speech Intelligent System Research Group, University of Southampton, (1998)
[11] A. J. Smola and B. Scholkopf. "A tutorial on support vector regression. NeuroCOLT Technical Report" NC-TR-98-030, Royal Holloway College, University of London, UK, (1998).
[12] V Vapnik. "The nature of statistical learning theory". Springer Verlag, New York, 1995.
[13] Marzia Polito, Pietro Perona, "Grouping and dimensionality reduction by locally linear embedding", Neural Information Processing System (2001)

LQ-learning with Self-Organizing Map for POMDP Environments

H. Y. Lee
Dept. of Elec. & Comm. Eng.
Tohoku Univ.
Sendai, 980-8579, Japan

H. Kamaya
Dept. of Elec. Eng.
Hachinohe Nat'l Col. of Tech.
Hachinohe, 039-1192, Japan

K. Abe
Dept. of Elec. & Comm. Eng.
Tohoku Univ.
Sendai, 980-8579, Japan

Abstract

In POMDP environments, the learning agent cannot observe the environment directly, thus partially observable states appeared. In order to overcome this partially observable problem, we have proposed a new RL algorithm, called "*Labeling Q-learning*", and further LQ-learning with SOM. By the LQ-learning with SOM, some knowledge about the environment is prepared ahead of learning process. The knowledge is automatically acquired by Kohenen's *Self-Organizing Map* (SOM).
In this paper, we examine what kinds of SOMs should be combined with LQ-learning over the same Grid-World problems.

1 Introduction

Reinforcement Learning (RL) is one of Machine Learning methods and an RL agent autonomously learns the action selection policy by interactions with its environment. Unfortunately in practical problems, the agent suffers from the incomplete perception: the agent observes the state of the environments, but these observations include incomplete information of the state. This problem is formally modeled by "*Partially Observable MDP*"(POMDP)[1].
One of the possible approaches to POMDP is using the historical information of state transition to estimate states. As a simple way to add memory to the agent for state representation, the agent defines a finite-width window of history including recent observations and, if needed, actions.
Another approach is to use a hierarchical RL algorithm. Wiering and Schmidhuber[2] had proposed the interesting method, called "*Hierarchical Q-learning*"(HQ-learning). In HQ-learning, a non-Markovian task is automatically decomposed into simpler subtasks solvable by usual memoryless Q-learning. However, hierarchical method maintaining the learning mechanism also in a simple problem, it is con-

sidered to be a thing with slight futility. Therefore, we proposed what is called "*Labeling Q-learning*"(LQ-learning)[3] previously, which is equipped with characteristic internal memory mechanism, called *Labeling* : the agent attaches a label to each observation. However, in application to the general POMDP problems, the landmark-like situations have to be predefined in advance. Indeed it may bar the application to the general maze problem of LQ-learning.
In order to solve this problem of predefinition, a new memory structure had been devised[4]. Unlike original LQ-learning, knowledge about the environment is prepared in advance. However, the knowledge just gives the information about a classification of state transitions. Further the knowledge is automatically acquired by Kohonen's *Self-Organizing Map* (SOM)[5]. The LQ-learning agent attaches labels to observations based on a map obtained by SOM. We were able to show the validity of LQ-learning equipped with SOM in the Grid-World problem. In this paper, we try to investigate how SOM is useful in LQ-learning, and which works better through experiments among three kinds of SOMs.

2 Labeling Q-learning

Here, we define the episodic (separated trial) learning task[2]. Namely, the agent is assumed to have a lifetime, T_{max}, separable into trials (or episodes). Each trial consists of at most S_{max} steps. Each trial ends in a special state called a goal "G", followed by a reset to a standard starting state "S".
In each trial, let o_t be an observation at step t. In LQ-learning process, an integer $\theta_t \in \Theta$ is attached to each observation o_t based on current and past observations, say historical memory, where $\Theta = \{0, 1, 2, \cdots, M-1\}$ (M: the number of labels). Thus, the LQ-learning agent gets an observation $\tilde{o}_t = (o_t, \theta_t)(\in \tilde{O} = O \times \Theta)$, as a new type of perception. It means that the LQ-learning agent gets the observation in its own way. Regarding the $\tilde{o}$ as a state relative, then the Q-values

on $\tilde{O} \times A$ (an finite set of actions) are updated by conventional Q-learning.

In LQ-learning, the agent attaches a new label when a landmark-like situation (i.e. noticeable state transition) occurs. Therefore it is very important to decide which state transition is decided as a landmark-like situation. In the previous LQ-learning, landmark-like situations are tuned by hand.

The landmark-like situation was predefined by the labeling condition parameter, *CHANGE* defined as follows.

Let $\sigma_t = o_1 o_2 \cdots o_{t-1} o_t$, a history (string) of current and past observations, and $\tilde{\sigma}_t = \tilde{o}_1 \tilde{o}_2 \cdots \tilde{o}_{t-1} \tilde{o}_t$, a history of current and past *labeled* observations. Then we define the Boolean variable *"CHANGE"* on this history space of observations. On the other hand, the labeling function (LF) h is defined as a function on the history space of labeled observations. Then, by using *CHANGE* and LF(indicated "h" in Eq.(1)), our labeling process is expressed simply below.

$$\begin{aligned} \theta(o_t) &\leftarrow h(\tilde{\sigma}_t) \quad \text{if } CHANGE \text{ is True,} \\ \theta(o_t) &\leftarrow \theta(o_t) \quad \text{if } CHANGE \text{ is False,} \quad (1) \\ \tilde{o}_t &\leftarrow (o_t, \theta(o_t)). \end{aligned}$$

In Eq.(1), the Boolean variable *CHANGE* and LFs are defined on the whole history from the initial observation at the starting state to the current observation. But we will, here, consider the case in which they are defined only on a fixed-width finite history, as stated below. Further, several initial observation of each trial, o_0, o_{-1}, o_{-2}, etc., may be necessary in order to define *CHANGE* and also h. In this case, they are supposed to be set equally with the observation at the starting state "S". This is similar to the case of labeled observations.

$\theta(o)$ is regarded as a function from the observation space O to the label space Θ. But the value $\theta(o)$ is initialized with a default value, say "0" at starting of each trial, and labels are renewed whenever *CHANGE* is true during a trial. This means that the value of $\theta(o_t)$ is the latest label attached to the observation $o = o_t$, before time step t.

Here we used one of LF, h.

$$\mathbf{LF} \quad h(\tilde{\sigma}_t) = min(M-1, \theta(o_t)+1). \quad (2)$$

Here M is the number of labels.

In practical problems, unfortunately, it is difficult to decide beforehand what kind of state transitions are considered as the noticeable events that serve as landmarks. In order to overcome this problem, it is necessary to find some methods of automatically acquiring

the landmark-like situation, and it should be solved by introducing a method which can classify situations of the environment. In the next section, we will propose a new idea of LQ-learning with Kohonen's SOM.

3 LQ-learning with SOM

As mentioned above, SOM is used in order to classify sequences of states transitions automatically. As shown in Fig.1 which shows the case of SOM_{sss}, each input datum consists of 3 consecutive observations. (input data for SOM are made on the basis of possible transitions in the Grid-1 shown in Fig.1). During making input data, all actions of agent are selected at random except reject action which means an action such as goes back to the last state. There is no action selection rule, therefore all input data are made from random moving of agent.

In previous paper, we were able to see that the SOM makes LQ-learning more efficient and widely applicable. Therefore, in this paper, we try to verify how the SOM combined with LQ-learning works well in the Grid-World problems. In order to investigate which SOM works well, three types of SOMs are compared here. One, SOM_{sss} is made based on the input data consisted in 3 consecutive scalar value of observations, and the second, SOM_{SASAS} is from 2 consecutive state transition, i.e. at step t, $o_{t-2} - a_{t-2} - o_{t-1} - a_{t-1} - o_t$. The last one is from 3 consecutive observations in binary representation.

All take 100 trials to make whole input data, and in addition, each trial consists of 1000 steps (1 step is equivalent to 1 state transition). SOM algorithm computes the input data so that they optimally describe the domain of observations. The winners are automatically organized as a meaningful 1-dimensional order in which similar winners are closer to each other in the grid than the more dissimilar ones. In SOM, we set the number of winner to "5", i.e., all input data are classified to 5 groups. In order to make SOM, we used *SOMPAK* which is available from the site of Kohonen's laboratory. In SOMPAK, the parameters are set to as follows;

- Initialization: The size of map is 5×1; neighborhood function is taken to be a step function.

- Learning:(the number in the parenthesis means the value used in the second learning) At the first learning, the number of times of learning is 1000(10000); the learning rate (notation of α in SOMPAK) is 0.05(0.02) (it decreases monotonically with the learning steps); the radius is 3(1).
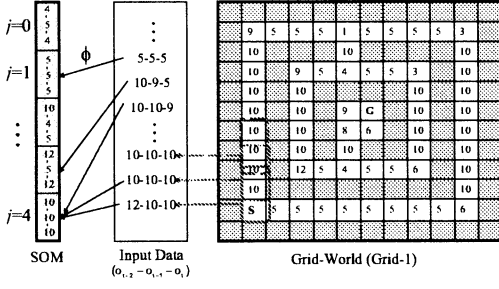
Figure 1: Example of how the SOM is made from Grid-1 (SOM_{sss}); Grid-1: 12×12 maze; The number indicated in each cell means the observation value. Once the agent reaches the goal, it receives a reward of 10. Otherwise the reward is zero; the numbers written in SOM means the winner of each group.

From SOMs, we can see that the state transitions which completely seem to be different belonged to the same group. There would be some relevance among the elements belonging to the same group. Unlike original LQ-learning, each group has a corresponding Q-table. Let ϕ be the mapping from the sequence of observations to the groups decided by corresponding SOM. For each sequence of observation with fixed length, the group, j is decided as follows;

$$j_{sss}(t) = \phi(o_{t-2}, o_{t-1}, o_t), \qquad (3)$$
$$j_{SASAS}(t) = \phi(o_{t-2}, a_{t-2}, o_{t-1}, a_{t-1}, o_t), \quad (4)$$
$$j_{bi-sss}(t) = \phi(bi\_o_{t-2}, bi\_o_{t-1}, bi\_o_t). \qquad (5)$$

Here bi_o_t stands for the representation of state-value in binary representation. In order to formulize a new type of observation including the information about group, j, let $\bar{o} = (o, j)$ be a new observation for new LQ-learning. Therefore the observation with label, $\tilde{o}$, is newly defined by $\tilde{o} = (\bar{o}, \theta)$. Then, the labeling functions for a new LQ-learning are defined as Eq.(6).

$$\text{LF} \quad \theta(\bar{o}_t) = \min(M - 1, \theta(\bar{o}_t) + 1). \qquad (6)$$

4 Experiments

4.1 Environments

We tested our LQ-learning on discretized 2-dimensional maze, called *Grid-World*. Grid-Worlds are not restricted in arrangement of obstacles, starting point, and goal point, therefore huge complexity may be realized. Fig.1 and Fig.2 present Grid-1 and Grid-2 used in experiments, respectively.

In each Grid-World, shaded cells mean obstacles, and

the agent can occupy any other cells in the maze. At each step, the agent can only see which of the adjacent cells are obstacles. The scalar observation value is computed by adding the value of shaded cells next to the agent, where the values of the northern, eastern, southern and western shaded cell are 1, 2, 4, and 8, respectively.
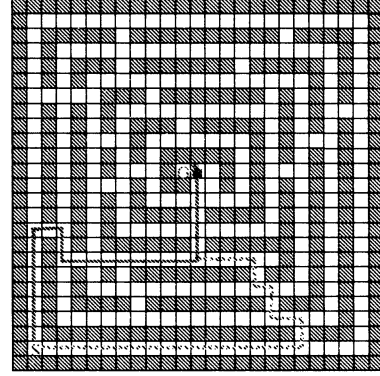
Figure 2: Grid-2; 25×25 maze; the optimal step is 27. The solid line shows the optimal route, and the dotted line means one of the actually learned routes, 38 steps

4.2 Action Selection

The agent has four actions: go north, go east, go south, and go west, except any action that would move the agent into a shaded cell. We further introduce "*reject action*", which means an action such as goes back to the last state.

The agent selects actions by ϵ-greedy rule that involves the random action selections. The value of epsilon is set to

$$\epsilon_T = (1 - \frac{T}{T_{max}})\epsilon_0, \qquad (7)$$

at Tth trial, where ϵ_T is the ϵ at trial T, and ϵ_0 is the maximum value of ϵ fixed at 0.1 in our simulations.

5 Results

Using three SOMs for classification of sequential observations or actions, a new LQ-learning agent is tested in two environments, Grid-1, and Grid-2. All results are presented by the average of 50 experiments.

5.1 LQ-learning in Grid-1

As shown in Table.1, all LQ-learning agents learn the near-optimal routes in Grid-1. Especially, all agents with SOM_{sss} perfectly learned the optimal route.

Table 1: Results of LQ-learning with 3 types of SOMs

	Grid-1		Grid-2	
	Av. steps	Goal(Optimal)(%)	Av. steps	Goal(Optimal)(%)
Without SOM	38.52	100.0(0.0)	56.34	58.0(0.0)
SOM_{sss}	27.00	100.0(100.0)	38.12	98.0(0.0)
SOM_{SASAS}	38.64	100.00(0.0)	38.18	88.0(0.0)
SOM_{bi-sss}	56.72	100.0(0.0)	38.79	96.0(0.0)
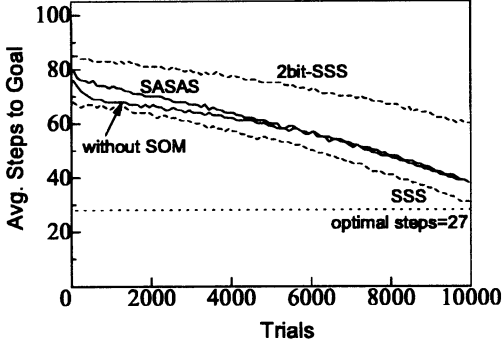
Figure 3: Result of LQ-learning in Grid-1: $T_{max} = 10000, S_{max} = 100$; The broken line shows the steps for optimal route.

Figure 4: Result of LQ-learning in Grid-2: The broken line shows the optimal path.

5.2 LQ-learning in Grid-2

In order to verify the applicability of SOMs, they are applied to a more extended and complex environment problem, Grid-2 shown in Fig.2. As shown in Fig.4, most of LQ-learning agents learn the near-optimal route only with the same learning time as Grid-1. However, in Grid-2, there are many data which don't belong to any groups of SOM, and then these are assumed to belong to the first group. In Grid-2, as similar as the results for Grid-1, more agents can learn the near-optimal route than ones without SOMs. From this result, we can say that devise of SOM makes LQ-learning more available in Grid-World problems.

6 Conclusion

In this paper, we investigated a new type of LQ-learning combined with Self-Organizing Map (SOM). In original LQ-learning, the predefinition of landmark-like situations may weaken the performance of LQ-learning in applying to general Grid-World problems. In the new LQ-learning with SOMs, it is not needed to predefine the landmark-like situation. Instead, the state transitions are classified by SOMs in advance.

Further we can see that LQ-learning agent with SOMs which were made from a distinct environment, work well in a rather big environment. By introducing the SOMs for preparation of initial knowledge of environment, it would be expected to be more widely adaptable for learning in unknown environments. Although the results are encouraging, learning in more complicated environments needs to be considered.

References

[1] R. S. Sutton and A. G. Barto, "Reinforcement learning : an introduction," Cambridge, MA: MIT Press, 1998.

[2] M. Wiering and J. Schmidhuber, "HQ-learning," Adaptive Behavior vol.6, no.2, 1997.

[3] H. Y. Lee, H. Kamaya, and K. Abe, "Labeling Q-learning in POMDP Environments," IE-ICE Trans. INF. & SYST., vol. E85-D, no. 9, pp. 1425–1432, 2002.

[4] H. Y. Lee, H. Kamaya, and K. Abe, "Labeling Q-learning with SOM," Proc. of Int. Conf. on Control, Automation, and Systems(ICCAS 2002), pp. 105–109, 2002

[5] T. Kohonen, "Self-Organizing Maps," Springer, 1995.

Behavior learning of autonomous agents in continuous state

Shon Min-Kyu*　　　　Junichi Murata*　　　　Kotaro Hirasawa†

*Graduate School of Information Science and Electrical Engineering, Kyushu University

Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8581, Japan

†Graduate School of Information, Production and Systems, Waseda University

2-2 Hibikino, Wakamatsu-ku, Kitakyushu-shi, Fukuoka 808-0135, Japan

e-mail: shon@cig.ees.kyushu-u.ac.jp

Abstract

This paper present a method for behavior learning of autonomous agent using modified Learning Vector Quantization (LVQ) in continuous state. Since the ordinary LVQ algorithm has a characteristic that learning speed is very quick, it can be used effectively in a reinforcement learning framework. However, since the ordinary LVQ network cannot know which is a good action among available actions, it sometime mis-learns due to erroneous teacher signals. The modified LVQ algorithm overcomes this problem. An example illustrates its validity in continuous space ploblems.

Keywords: Learning Vector Quantization, Reinforcement Learning, Continuous State, Autonomous Agent, Path Finding.

1 Introduction

Reinforcement learning[1] has been widely used in behavior learning of autonomous agents in unknown environments. An agent learns its optimal behaviors by trial and error in an environment. Reinforcement learning methods such as Q-learning[2] algorithm assume that the environment is a discrete space and that the agent learns the mapping from states to actions.

In our recent research we introduced a modified LVQ algorithm that is faster when used in reinforcement learning framework than Q-learning algorithm in discrete environment because the LVQ network maps the input information directly to the best action[3].

The LVQ networks[4][5] have weight vectors each of which has a label (output value of output node). They are trained by supervised learning algorithms. In this paper, the labels correspond to the actions. During the training process, LVQ network selects an action, and a weight vector corresponding to selected action is updated. However, since ordinary LVQ network cannot know how much the selected action among available actions is good, it sometimes mis-learns in the reinforcement learning environment.

Here a new LVQ algorithm is proposed to overcome this problem. The modified LVQ algorithm imposes a limit on how close a weight vector can approach an input vector. This limit is realized by changing the learning rate appropriately as the learning proceeds, and the limit depends on the amount of the reward that the action gets. Therefore, other weight vectors cannot be closer to the input vector.

However, environments in most of real world problems are continuous spaces. Therefore, the agent must learn the mapping between continuous valued state space and actions in order to deal with continuous space.

Here, we propose use of the modified LVQ algorithm in continuous space problems.

An example is provided to show the effectiveness of the proposed method.

2 LVQ Algorithm

2.1 Ordinary LVQ algorithm

Kohonen's LVQ is a supervised learning algorithm associated with the competitive network. A weight vector is associated with each node in the output layer.

The LVQ network calculates Euclidean distance d between the input vector $x=(x_1, x_2, \cdots, x_p)$ and weight vector $w_j=(w_{j1}, \cdots, w_{jp})$, $(j=1, \cdots, c)$ as in Eq.(1),

$$d_j = \sqrt{\sum_{i=1}^{p} (x_i - w_{ji})^2}. \qquad (1)$$

Then, the LVQ network selects a weight vector w_k

('winner') which minimizes distance d, and the selected weight vector $\boldsymbol{w}_k$ is trained as follows:

$$\boldsymbol{w}_k := \boldsymbol{w}_k + \Delta\boldsymbol{w}_k,$$
$$\Delta\boldsymbol{w}_k = \pm\eta(\boldsymbol{x} - \boldsymbol{w}_k), \qquad (2)$$

where $\eta > 0$ is the learning rate. When the label value of the winner matches the output value of training data, the plus sign in Eq.(2) is adopted, and otherwise, the minus sign is used.

After learning, the LVQ network chooses the nearest weight vector to a given input vector, and outputs the label corresponding to the selected weight vector as the network output.

In the case of the ordinary LVQ algorithm, for a given training pair, its corresponding winner weight vector approaches the sample input vector, and if the same training pair is given repeatedly, the weight will approach the sample input vector without limit of distance.

2.2 Mis-learning in the reinforcement learning framework

Since Kohonen's LVQ algorithm is a supervised learning algorithm associated with the competitive network, the weight vectors are trained by teacher signal. However, the reinforcement learning is unsupervised learning. Thus the LVQ network cannot know correct output vector or optimal action in the reinforcement learning framework. In this paper, the LVQ network carries out the role which selects an action for given input information and updates the weight vector of output node corresponding to the selected action. However, during the training process, it cannot be known which is the good action. The LVQ network learns only the action which gets the reward.

For example, assume that there are three available actions which can be chosen. If an action gets punishment, the LVQ network can know it is wrong. However, other actions which lead the agent to the goal can get a reward. An action 'go-right' may eventually leads to the goal and thus may get a reward. Therefore, the weight vector corresponding to 'go-right' may be updated to approach the current input vector by the LVQ algorithm. However another action 'go-forward' might have been the best action in the sense that it takes the agent to the goal most quickly. Once the 'go-right' is selected and its weight vector is update to approach the input vector, the same action becomes more likely to be selected. Therefore, the action 'go-right' will be repeatedly selected, and the weight vector associated with this sub-optimal action can be closer to the input

vector than the optimal action 'go-forward'. Eventually, the learning agent may learn not the optimal action but the sub-optimal action.

A new LVQ algorithm must be devised which can deal with these erroneous and incomplete training data in the reinforcement learning framework.

3 Modified LVQ Algorithm

The modified LVQ algorithm can overcome the above problem. The modified LVQ algorithm imposes a limit on how close a weight vector can approach an input vector. This limit is realized by changing the learning rate appropriately as the learning proceeds, and the limit depends on the amount of the reward that the action gets. Therefore, an incorrect weight vector can not be closer to the input vector than the correct weight vector.

The modified learning algorithm defined as follows:

$$\boldsymbol{w}_{(k+1)} = \boldsymbol{w}_{(k)} + \alpha_{(k)}s_{(k)}r_{(k)}(\boldsymbol{x}_{(k)} - \boldsymbol{w}_{(k)}), \qquad (3)$$

$$\alpha_{(k+1)} = \frac{\lambda\alpha_{(k)}r_{(k)}}{(1 + \lambda s_{(k+1)}r_{(k)}\alpha_{(k)})r_{(k+1)}}, \qquad (4)$$

where,

k	:	iteration number,
$\alpha_{(k)}$	:	learning rate,
$r_{(k)}$	:	amount of reward,
λ	:	a constant, $0 < \lambda < 1$,
$s_{(k)}$	:	$\begin{cases} +1 & : \text{if reward is received,} \\ -1 & : \text{if punishment is received.} \end{cases}$

Due to Eq.(4), the contribution of $\boldsymbol{x}_{(k)}$ to the weight update will be as λ^k times strong as that of $\boldsymbol{x}_{(0)}$ for any k. In other words, the contribution decays exponentially as the learning proceeds.

If $r_{(0)} = 1$ and $\lambda = 1$, Eq.(3) and (4) are reduced to Kohonen's OLVQ1 algorithm. Now, let us consider a situation where the algorithm accepts the same input vector $\boldsymbol{x}$ repeatedly. By letting $\boldsymbol{x}_{(k)} = \boldsymbol{x}$, $r_{(k)} = r$ and $s_{(k)} = s$, we get

$$\boldsymbol{w}_{(k)} = \frac{1 - rs\alpha_{(0)}}{1 + sr\alpha_{(0)}\sum_{j=1}^{k-1}\lambda^j}\boldsymbol{w}_{((0))} + \frac{rs\alpha_{(0)}\sum_{j=1}^{k-1}\lambda^j}{1 + sr\alpha_{(0)}\sum_{j=1}^{k-1}\lambda^j}\boldsymbol{x}. \qquad (5)$$

Thus, since $0 < \lambda < 1$,

$$\lim_{k \to \infty} \boldsymbol{w}_{(k)} = \frac{(1-\lambda)(1-rs\alpha_{(0)})}{1-\lambda+rs\alpha_{(0)}\lambda}\boldsymbol{w}_{(0)} + \frac{rs\alpha_{(0)}}{1-\lambda+rs\alpha_{(0)}\lambda}\boldsymbol{x}. \tag{6}$$

From Eq.(6), we can represent the distance between the final weight vector and the input vector in terms of the distance between the initial weight vector and the input vector

$$\boldsymbol{x} - \lim_{k \to \infty} \boldsymbol{w}_k = \frac{1-\lambda}{\frac{1}{1-rs\alpha_{(0)}} - \lambda}(\boldsymbol{x} - \boldsymbol{w}_{(0)}), \tag{7}$$

where

$$d = \frac{1-\lambda}{\frac{1}{1-rs\alpha_{(0)}} - \lambda}. \tag{8}$$

Therefore, even if the same input vector is repeatedly input to the algorithm, the weight vector cannot approach the input vector indefinitely. There is a limit on how close the weight vector can approach the input vector thanks to the learning rate Eq.(4).

The limit d depends on the amount of reward assigned to the input vector $\boldsymbol{x}$. When $s = 1$ and reward is big, d will be small. As the received reward becomes larger, the weight vector can be closer to the input vector. Therefore this modified LVQ algorithm can solve the problems described in the previous section. See [3] for more details.

4 Example

The modified LVQ network (Fig.1) has been applied to a simple behavior learning problem in order to certify its performance in continuous state space. The goal of this example is to find a path from the starting point to the goal area avoiding four obstacles in the environment (30cm x 50cm) as shown Fig. 2.

The agent (3cm x 3cm) has three sensors whose readings are fed to the LVQ netwrok as inputs x_0, $\cdots$, x_5, (left(x_0, x_1), front(x_2, x_3), and right(x_4, x_5)). Each sensor can detect angle ($0° < x_1, x_3, x_5 < 180°$) and distance ($0\text{cm} < x_0, x_2, x_4 < 5\text{cm}$) of the nearest obstacle as real values (Fig. 3). The agent can choose its action among 'go-forward', 'turn-right and go-forward', 'turn-left and go-forward' and 'turn-backward and go-forward' which are outputs of the LVQ netrork. Here, in one time step, the agent can move up to 3cm. But if there is an obstacle in its way, it stops just before the obstacle. The resolution of the agent's movement is 0.1mm.
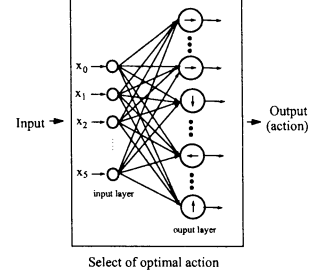
Figure 1: Structure of modified LVQ network.
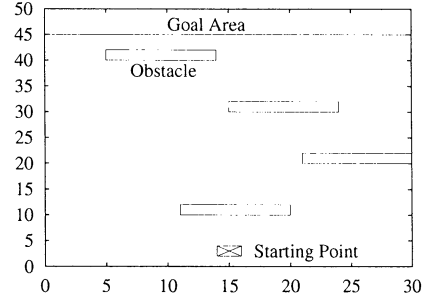
Figure 2: Environment.

During exploring the environment, the agent selects its action based on ϵ-greedy strategy[1]. It does not always select the greedy action but selects other action at a probability of ϵ, and performs the selected action. Therefore, the agent can explore more widely by this method in the environment.

We update the weight vectors after every episode. An episode is a period during which the agent sets off the start point and ends in a terminal state. There are two terminal states. One is the state where the agent reaches the goal area, and the other is the time-up state when the agent cannot reach the goal area within a prespecified number of steps $N = 200$. When the agent has successfully arrived at the goal area, the weight vectors of the LVQ output nodes activated along the path to the goal area are updated by Eq.(9),(10)

$$\boldsymbol{w}_{(k+1)} = \boldsymbol{w}_{(k)} + \alpha_{(k)}r_{(k)}s(\boldsymbol{x}_{(k)} - \boldsymbol{w}_{(k)}), \tag{9}$$

$$\begin{aligned} s &= 1, \\ r_{(k)} &= 1 - \frac{total\_step_{(k)}}{N}, \end{aligned} \tag{10}$$

front

5cm

sensor
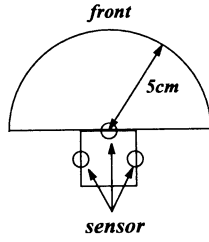
Figure 3: Structure of an agent.

Table 1. The number of steps of the best path.

Random number sequence	LVQ algorithm	Proposed LVQ algorithm
1	Failed	20 steps
2	Failed	24 steps
3	Failed	26 steps
4	30 steps	22 steps
5	Filed	20 steps

where k is the episode number, $\alpha_{(k)}$ is calculated by Eq.(4), and $total\_step_{(k)}$ is the number of moves that the agent has made in this episode k. The reward $r_{(k)}$ is equally given to all the actions in this episode. If $total\_step_{(k)}$ is small, i.e. the agent finds the goal area quickly, the agent gets a large reward $r_{(k)}$. This, combined with the modified LVQ, drives the agent to find more good action. When the agent cannot reach the goal area within a prespecified number of steps N, the reward $r_{(k)}$ and its sign s are set as

$$
\begin{aligned}
\alpha_{(k)} &= 1, \\
r_{(k)} &= 0.00005, \qquad (11) \\
s &= -1.
\end{aligned}
$$

Training was done for 500 episodes with 5 different randomly initialized weights.

Table 1 shows the length of route (measured in the time step) that was found by the Kohonen's LVQ algorithm and the modified LVQ algorithm. In Table 1, the ordinary LVQ algorithm can hardly find the route to the goal area, while the modified LVQ algorithm can in all cases successfully find the route to the goal area.

Since the ordinary LVQ algorithm blindedly updates all actions which get reward, it does not know which action is actually good. However, since the

modified LVQ algorithm knows how much a selected action is good and updates its weight vector accordingly, it easily finds the route to the goal area at a high probability.

5 Conclusions

This paper proposed use of modified Learning Vector Quantization (LVQ) for behavior learning of autonomous agent in continuous state. As an example, the proposed method has been applied to find a better action in continuous environment which have many routes from the start point to the goal area.

In order to prevent mis-learning in reinforcement learning framework, the proposed method limits the distance that a weight vector can be close to input vector. From the results of the example problem, it has been found that the modified LVQ performs much better than the ordinary LVQ. This certifies the validity and usefulness of the proposed LVQ-based reinforcement learning method.

However, since movement of the agent is discretized the output space is not continuous. In our another research, we also introduced a new LVQ algorithm which can output continuous values with fuzzy algorithm[6].

As a future subject, we will combine this method and the modified LVQ, and make an algorithm which can be applied to purely continuous problems.

References

[1] Sutton, R. S. and A. G. Barto, Reinforcement Learning: An Introduction, MIT Press, 1998.

[2] Watkins, C.J.C.H.and Dynan.P., "Techmical note : Q-Learning" *Machine Learning*, 1992.

[3] Shon. M.K., J. Murata and K. Hirasawa, "Behavior learning of Autonomous Robots by Modified Learning Vector Quantization", *Trans. of the Society of Instrument and Control Engineers*, 2001.

[4] Kohonen. T, "Self-Organizing Maps", *Springer Series in Information Sciences*, 1995

[5] Kohonen. T, "Learning Vector Quantization for Pattern Recognition", *Technical Report TKK-F-A602*, 1990

[6] Shon. M.K., J. Murata and K. Hirasawa, "Function Approximation using LVQ and Fuzzy sets", *Proc. of Int. Symp. on IEEE SMC*, 2001.

2 － 1 － 2 1 Hagiwara, Oita, 870-0921, Japan

Tel: 097-551-8148, Fax 097-552-0360

E-mai: info@shubundo-p.co.jp

http://www.shubundo-p.co.jp